Huxor, Avon Paolo, 1994.
Artificial intelligence as writing: knowledge-based hypertext systems as a medium for communication.

Artificial Intelligence As Writing: knowledge-based hypertext systems as a medium for communication

Avon Paolo Huxor

A thesis submitted in partial fulfillment of the requirements for the degree of Doctor of Philosophy

April 1994

Middlesex University

Abstract

This thesis is an exploration of a new metaphor for artificial intelligence (AI). Traditionally, the computer within AI has been viewed as an agent, one with which the user engages in a conversation. More recently certain researchers have proposed the notion that artificial intelligence (and indeed computing in general) can be more appropriately seen as a form of writing.

Initially this thesis reviews the literature in this area, and aspects of AI which support the approach. Features of writing are then described which show parallels with AI. This then allows us to take lessons from the history and development of both traditional writing and the new computer-based writing systems to inform the design of a new type of artificial intelligence system.

A design based on these features, called Running Texts is presented through a number of small examples. Issues that arise from these and possible future developments, based on the implementation are then discussed. A rationale for users choosing to learn a system such as Running Texts is proposed, as benefits from the psychological and social implications of writing can be applied to AI systems, when they are seen as writing. The same parallels point out potential problems, and suggest new ways to see the relation between AI and thought..

Table of Contents

List of Figures

List of Tables

Introduction

This thesis explores in some detail the idea that work in artificial intelligence (AI) can be seen in a new way, as a form of writing — a new type of medium for communication between people. This idea contrasts with the more traditional view of AI which sees it as the science and technology of creating autonomous, intelligent, machines through an understanding of human intelligence. Within the literature this traditional view of AI has been formalized into two goals that help define it (Hill 1989):
- to understand the human mind by creating computational models, and
- to build computer systems that perform actions that are conventionally regarded as intelligent.

In the design of intelligent machines, the dominant metaphor has been that of the computer as an *agent*, that is, the computer is another 'person', one that can engage the user in a dialogue. The ubiquitous nature of this metaphor is partly due to the human inclination to conversation, and leading from that, the particular influence of the Turing Test on AI, a test in which the computer must prove its worth by passing as a conversational agent, its behaviour indistinguishable from human conversational performance. The very name 'artificial intelligence' indicates that the original view of the subject is one of creating intelligent machines, and only more recently have researchers and departments moved to use other terms to describe their work: terms such as knowledge-based systems, which do not carry the same connotations.

More recently, however, there has been a sea change in approaches to AI. Some researchers (especially Bolter 1991) have posited that computing in general can be viewed as a form of writing. Related to this proposal others, such as Sokolowski (1988), are more specific, arguing that there are significant similarities between artificial intelligence and writing. Moreover, these similarities are not just chance coincidences, and our understanding of writing can throw light on the nature of AI, its possible applications and effects. Hill (1989) also exemplifies this new and developing approach to artificial intelligence, and he identifies a third goal for AI, in addition to (but not necessarily replacing) the two described above:

> This third goal is establishing new computation-based media in which human intellect can come to express itself with a different clarity and force (1989: 30).

Hill believes that AI techniques fall into the same category of representational media as writing, photography and photographic composition, cinema and cinematic convention. That is, they should be seen more as "representational products of mind than as direct representations of mind" (1989: 31). This is a view equally shared by Anderson & Holmqvist (1990) who see two possible ways of approaching AI:

- The ontological: which argues that there are non-trivial similarities between machine and human, and one can learn about one by investigating the other. This approach combines the first two goals of traditional AI described above.
- The pragmatic: in which AI techniques provide useful and interesting interfaces, but are, in principle, illusions created by the skillful exploitation of aesthetic techniques. This is essentially Hill's third goal.

In their own work Anderson and Holmqvist adopt the latter, pragmatic, view: for them computer systems are more usefully seen as a medium. AI techniques, in this view, are aesthetic techniques for producing kinds of signs that convey specific information or emotions to the reader. In other words, they can specify the roles (not real minds) and staged actions and emotions (rather than real actions and emotions) that allow AI to create real experience and emotions in the real human audience. The techniques of AI are "a special type of computer based sign" (1990: 4).

It is this approach that this thesis explores and extends, seeing existing AI systems as representations of the thoughts of the systems' authors, rather than as being independent thinking systems themselves, and proposing designs for future AI systems that explicitly exploit this position. It will investigate other representation systems, primarily writing, but also, to a lesser degree, other symbol systems such as musical notation. These existing media exhibit a number of parallels with the work in AI, and by extending these parallels further, this thesis suggests a design for a new type of artificial intelligence system, and indicates some of the issues that might arise in its use.

The Thesis

The work described in this thesis has its origins in research that the author began in 1986 (Huxor 1986), which investigated how the techniques of artificial intelligence might be employed as a means for the generation and communication of knowledge, that is, as the basis for a *knowledge industry*, rather than as a means to simulate human cognitive performance. At that time the specific means for using the computer for the communication of knowledge between human users was unclear. Like others I initially worked with the metaphor of a knowledge refinery — as if knowledge engineering were an industrial process whose product was knowledge (see Michie & Johnston 1985). Knowledge is not a consumer product, but is intimately associated with understanding. Much as the aim of using the technology to create knowledge archives, improving the quality of the material and returning this knowledge back into the human domain seemed a reasonable one, a specific mechanism to communicate the knowledge to potential users was required. However, problems have been found to occur in user acceptance of computer-mediated knowledge if the communication takes place through the traditional AI mechanism in which the machine is a tutor to an ignorant human, problems that are discussed in detail in chapter 2. An alternative approach is required.

A more specific motivation for the work described herein was the difficulties encountered in applying the conventional AI approach to the design of a decision aid for economics (Huxor 1988). The domain of economics is one that is controversial — economists often disagree. The traditional AI approach, however, was based on the assumption that one can find an expert whose knowledge is sufficient to be placed in the machine. The user then engages in a dialogue with this 'expert' system to obtain answers to the questions that he or she may have, answers that are 'right'. But the fact that the system embodies an ideology, that of its designer and implementer, is lost, and, as already stated, this style of human-computer interaction, with the supposedly unknowing user obtaining the correct answers from the machine, poses problems. This problem has even arisen in one of the more famous expert systems, Prospector, which "was unable to take into account disagreements among experts, the full significance of which was not reported until towards the end of the project" (Clark 1991: 20). Early evaluations only compared the match between the system's output and that of the (one) expert who designed the model implemented. But if there are problems with the conventional methodology, what alternatives exist to the machine as 'oracle'? One solution that emerges is that of the computer as a medium, although a very dynamic medium, to facilitate the communication of ideas between people — rather than AI as oracle or industrial process.

In a wider context, the ideas presented can be considered as part of a confluence of recent research in both artificial intelligence and writing. There is, above all and as briefly mentioned above, growing concern within the AI community about the directions it has been going, and alternatives are actively being sought. Secondly, the role of writing as a tool of the intellect has brought the issue of the relation between any form of writing (or notation) and human reasoning to the fore, and brought issues from the contemporary study of writing into the traditional domain of artificial intelligence.

This thesis takes these ideas further, and the particular claims are that:

• Artificial Intelligence, although currently based on the machine as agent, can be re-interpreted either as a reflective tool or, more usefully, as a communication medium. That is, AI (in its symbolic form) can be seen as a form of writing in its broadest sense: an external and concrete notation of an existing practice. AI can, furthermore, be located within recent developments of computer-based writing — knowledge-based and dynamic hypertexts. This notion of 'AI as writing' can also be employed as a basis for discussing the potential long-term implications of AI.
• The arguments above can be used to inform the design of a new type of knowledge-based system that facilitates the expression and communication of knowledge between people. There is sufficient material for one to be specific about a working design.

The issues that arise in this study range over both theoretical and implementational concerns. Abstract issues in AI and writing are shown to share common concerns, such as the role of conventionalization in a representation, and phonocentrism (the view which places speech above writing). On the practical side, existing implementations within AI and computer-based text systems can — at a deeper level — be shown to be moving in three directions which are bringing them together:
Firstly, recent AI systems have tried to overcome the problems that come from being an 'expert in a box', by introducing new interface metaphors, and moving the locus of control to the user.
Secondly, computer-based writing systems, initially built to mimic existing paper-based text systems (such as the typewriter) are increasingly exploiting the computational nature of the machine in which they are implemented to provide additional, dynamic, functionality.

And thirdly, throughout its development, writing has shown a tendency to allow the author to add structure to assist the reading of the text. Computer-based hypertexts have encouraged this and taken the notion further by allowing the structures to be dynamic, rather than static, objects.

Putting these three strands together suggests design features for the implementation of a knowledge-based medium, which I have called *Running Texts*. In Running Texts, the techniques of AI are imported into text documents, much as a spreadsheet incorporates simple algebraic equations. Furthermore, these units of AI code can also be considered as a means to express the rhetorical structure of the text, connecting paragraphs from outside (giving them structure), rather than only being a few sentences within them. This integration enables the creation of a medium that represents and communicates what can be thought, in much the same way that conventional writing represents and communicates what can be said. One can create a tool to 'write down what one is thinking'. In practice, using a Running Text is similar to any writing task, in that the author assembles a series of AI techniques into a document, much as one assembles a series of words to write a sentence. The reader can both read the text, and manipulate its contents through citation and modification so as to create a response to the text.

Treating AI as a kind of writing has a number of other potential consequences which are addressed in later chapters. AI has always had a concern for the social and cultural aspects of its ideas and products (see for example Boden 1987: part VI, and Boden 1990). Similarly the new view allows us to propose a new set of benefits and concerns for the technology.

Thus, the thesis chapters will proceed as follows:

- First, a discussion of traditional artificial intelligence, its aims, methods and problems is presented to give a context for the thesis. (Chapter 1)
- Secondly, an investigation of the perceived role of the user of AI systems, and the importance of dialogue in AI in particular. Alternative interface suggestions are discussed, leading towards the notion of a medium. (Chapter 2)
- Thirdly, a detailed discussion of the established technologies of the word: writing and print, is undertaken to explore the parallels between writing and AI. (Chapter 3)
- We then examine developments in computer-based writing systems, which

have gone beyond simple word-processing systems and incorporate the computational features of the machine. (Chapter 4)

• There follows an exploration of developments in computer-based text systems in which the structure of the text is explicitly represented, a structure that can be computationally derived in certain situations. (Chapter 5)

• This theoretical and practical background material is then drawn together to inform the design of a new knowledge-based writing system, whose implementation, known as the *Running Text* system, is described. (Chapter 6)

• This is followed by a discussion of the uses and issues involved in the Running Text system, through the use of various examples of use. (Chapter 7)

• There follows a discussion of the psychological and social implication of AI technology. By investigating the effects of writing, we can metaphorically suggest some of the benefits and problems that employing 'AI as writing' might involve. (Chapter 8)

• Finally, concluding remarks are made. (Chapter 9)

1. The Nature of Artificial Intelligence

As a distinct academic field with its own conferences and journals, AI has a relatively short history, going back only some forty years, and its birth can be traced quite clearly to a meeting in the mid-fifties at Dartmouth College, USA. Of course, its intellectual precursors reach back much further into history. Indeed, the nature of what really are AI's precursors is a major theme of this thesis, which argues that the history of the word offers more than the history of the automaton, to which AI has traditionally turned.

It is important to look at those ideas which AI itself considers to be its historical and cultural context, because they give us an indication as to the motivations and long-term goals that underlie the project. Popular books on AI (for example, Feigenbaum & McCorduck 1984) refer to a number of wonderful devices and beasts: the robot-like 'attendants to Hephaestus' of the *Iliad*, who were "in appearance like living women/There is intelligence in their hearts, and there is speech in them" (McCorduck 1988: 69). In the same period, King Mu of China is said to have had an immortal robot which was very close to having artificial flesh and blood. Later times gave us the Golem created by the Rabbi of Prague, and the monster forged by Mary Shelley's Frankenstein. On a less dramatic scale, but more closely aligned to the goals of AI, from medieval Europe came reports that Pope Sylvester II, Albert Magnus and Roger Bacon (amongst others) were said to have fabricated talking heads of brass that could predict the future and solve difficult problems (McCorduck 1988).

It is certainly curious that a subject which claims to be scientific looks so proudly to such origins and such an approach leads one to wonder why many AI people are surprised at public distrust. Although no serious researcher would openly claim to use these devices as inspiration, they do still lie in the background of much current work.

What are the stated aims of researchers in the field, when one returns to the reality of contemporary work? Barr and Feigenbaum, in one of the definitive classics of AI, state that : "Artificial Intelligence is the part of computer science concerned with designing intelligent computer systems" (1981/2: 3). Even more strongly, Woody Bledsoe, in his presidential address to the American Association for Artificial Intelligence (AAAI), when asked why the members of the association undertook AI research, assessed that the basic reason for undertaking AI was "a yearning we have to make machines act in some fundamental way like people" (1986: 57). He then went on to list his dreams: Buck Rogers, HAL, 2010, Star Wars, R2D2, CP30 and the Turing Test. More recently, and despite evidence of problems (or perhaps due to these) a call has gone out to stick to these original aims:

> Abandoned as either impossible or unattainable, the real goal of AI, the creation of an intelligent machine, has been left aside for too long (Monterege, 1989).

Of course, few researchers or commercial companies aim to achieve this goal; in the short-term more practical and useful results are required. But the problem is that this dominant goal of AI guides the general direction of development, and the potential for the technology to develop in other directions are not fully exploited.

But progress has been made within the field. This is substantially due to the fact that AI, of course, consists of many parts: not only a set of long-term goals, but also a set of techniques, and a number of working programs. The true nature of AI depends on which of these is considered as the most important. It is significant, that despite their attachment to the ideology of AI, Barr and Feigenbaum note that their book, *The Handbook of Artificial Intelligence*, a classic in the field,

> is an encyclopaedia of the major developments of the field's first 25 years - programs, programming techniques, and the computational concepts used to described them (198/2: 3).

Their handbook seems less concerned with issues of creating autonomous systems, and more with the practical issues of programming tools and techniques. It is this attitude to AI that this thesis adopts. That is, the programming techniques and computational concepts developed over the past thirty years can be employed in the service of another goal. For this reason, although I question the goal of designing autonomous and intelligent machines, and seek to design a knowledge medium, the work described can still come under the umbrella term of AI, as it draws upon the valuable techniques that AI has developed. Furthermore, the close relationship between writing and thought discussed in this thesis helps throw light on many of the issues that are unique to AI. Although it may not follow the goals of traditional AI, the new approach still addresses many of the issues inherent in them. In fact, one can go further. Speculation on the relationship between writing and thought indicates that one reason why AI actually succeeds in 'simulation' is that it takes writing and makes it dynamic. But writing has been interiorized by humans over time, the external tool being brought into the cognitive realm, and thus aspects of human cognitive performance, those drawn from use of conventional writing, can be apparently simulated using the computer as a means to activate the written word.

AI and metaphor

The traditional approach to AI described above is a metaphorical one. The machine is seen as a partner, similar to a human partner. It is an anthropomorphic approach that "looks to the world of human-human communication for inspiration about the nature of human-computer communication and action" (Waterworth & Chignell 1989: 213). One of the most important aspects in the development of AI has been the role of metaphors such as this. This is, as West and Travis (1991a) argue, due to the youth of AI as an area of study, it necessarily lies on Quine's "fringe of science", where metaphor has a major role in shaping theory and guiding research. The arguments presented below are based on the belief that the idea of the computer as a human is itself a metaphor and can, therefore, be replaced by another, namely one that sees parallels with writing. West and Travis further note that metaphor has a significant role not only in providing a framework for explaining results, but also in guiding further research. In this role the value of any metaphor can be judged by its fecundity, its ability to suggest interesting and useful directions for research, as well as explaining what already exists. I hope that the work below indicates the usefulness of the 'AI as writing' metaphor, which both provides a means to reinterpret a wide range of existing issues within AI, and indicates a number of new solutions for system design.

The issue of metaphor is, however, not a simple as it might initially appear to be. Travis and West (1991a) propose the term *paraphor* to denote a metaphor whose use is so common that its metaphoric nature is forgotten. Their concern is mainly with seeking new metaphors for mind, rather than new metaphors for AI, but similar arguments apply. It can be maintained that the metaphoric nature of AI as 'mind' has become paraphoric through extensive use. It has become difficult to imagine AI technology in any other way, and recent research that has sought to propose alternatives has often had to struggle to get accepted. Naturally, the same arguments can be used to critique the ideas laid out below. If one argues that AI is writing, could not the same 'error' be made? Should we just employ the metaphor to explain certain results and assist in design, avoiding the move that takes us from proposing that we can see 'AI as writing', to arguing that 'AI is writing'. We must ask ourselves whether 'AI as writing' is another metaphor, or whether the description of writing can (or should) be broadened to include aspects of the results of AI. Although this thesis was begun with the intention of using the notion as a metaphorical guide, findings from the study of writing suggest that extending the definition of writing to include AI is not inappropriate. This latter approach has been taken by theorists of writing, the deconstructionists in particular, who define writing in a much broader sense than we are used to, meaning something like "making public" in any manner (Bruffee 1984: 641), which could easily encompass representations encoded into a computer. Of course, it should be noted that a similar strategy, of broadening definitions, has also been taken by some traditional AI workers, many of whom take the intellectual core of AI not to be the building of intelligent machines, or the simulation of human cognitive performance, but the idea that AI is the science of intelligence in general as this is expressed in any system, biological or otherwise (Boden 1990: 1). The use of the term 'intelligence' is thus been expanded to include reasoning by machine as well as by humans. Within this thesis both uses of the term writing are employed: as straightforward metaphor, and as genuinely inclusive of AI. Often it is difficult to distinguish the one from the other.

Before asking whether we should seek an alternative to the 'AI as agent' metaphor, it is useful to address, briefly, the question of why it became so ubiquitous. There are a number of reasons why the 'machine as agent' metaphor should have come to the fore. Firstly, there is the human desire to anthropomorphise, to project human qualities on whatever lies before us. Walt Disney built a commercial empire by exploiting this very desire. There is also, however, another more technical issue. Initially, the design of human-computer interfaces was seen as the design of dialogue. This was encouraged by the primitive technology of the time, which being a simple teletype could either accept input typed sentences or output printed replies. But the increasing power of the hardware, which enables the production of complex graphical displays, has allowed interface designers to use other metaphors such as that of the desktop or notebook. The idea of a dialogue has been dropped to a large extent by those who write software (with the notable exception of AI software). The user no longer engages in a form of discourse with the machine, but moves and manipulates graphic representations of everyday objects. The Apple Macintosh, on which these words are being written, is a very good example of this development.

Another contributory factor that explains the commercial support for a considerable amount of AI research can be attributed to the promise that it offered manufacturers of complex computer systems. The widespread acceptance of computer technology into offices and homes has been slowed down by the difficulty users have in learning and using them. If AI technology were to deliver on its promises, the user could ask the machine in plain language to undertake whatever task was required, and any response would equally be in everyday dialogue. Natural language database query systems are a product of this sort of investment. It is almost a given that the role of human-computer interface design is to make the systems' facilities available to everyone. The user conversing with the machine in their natural language seemed the most desirable solution.

Due to its stated aims, AI is the last bastion of the anthropomorphic principle, but the application of other metaphors can, arguably, assist in increasing the usability and success of knowledge-based systems. The idea that the computer could converse with the user may appear the most likely to aid them, but the success of the Macintosh, with its graphic interface, suggests that other approaches should be investigated. As Soloway has succinctly put it, the primary metaphor for all AI research has been that "of one HUMAN entity interacting with another HUMAN entity". But he asks:

> How do I want that intelligent notebook...to function? What unique non-human features can/should be exploited (1991: 131).

This thesis answers exactly this point. The features that are proposed are those of the many *technologies of the word*: writing, print and hypertext. There are disadvantages to the solution proposed: in particular the users must learn many of the complexities of the system just as one learns any writing system. But the fact that we bother to learn to read, rather than ask others to read out things for us, indicates that there are perceived benefits for the individual.

Writing and intelligence

At the same time that AI is beginning to consider itself in new ways, and ways that bring it towards writing, those engaged in the study of writing are returning the compliment. There are two forms that one can discern; one applies methods used by AI to the analysis of specific literary texts, and the other is a more general theoretical reappraisal of writing which is concerned with its representational capabilities.

In the first case, there is an increasing use by literary analysts of the knowledge elicitation techniques created initially by those intending to analyse and implement AI systems. Typically, a researcher in AI will take a protocol, that is a written version, of spoken interaction (see Ericsson & Simon 1984), or indeed simply turn to existing written material. The protocol (or text) is used as source data for the construction of a cognitive model which hopefully explains the data. A similar close analysis, but applied to literary texts has been successfully applied to uncovering the cognitive aspects of metaphor in such writers as Milton (Turner 1987) and the poems of Robert Frost (Holland 1988), in what Holland calls a 'cognitive approach to literature'.

In the second, and more important, case, recent years have also seen a questioning of the view that goes back to Plato and Aristotle, one that finally became a commonplace in Bloomfield's dictum— that 'Writing is not language, but merely a way of recording language by means of visible marks'. Thus, writing became a second class citizen, the sign of the sign. This priority of speech over writing can be traced, according to Derrida (1967/76), to a deep phonocentrism that lies in Western thought. He argues that although we may believe that hearing ourselves speak makes us fully present to ourselves, in fact, even when we speak, either openly, or internally (i.e. think), we still employ a systems of signs; and a system that is not ours alone but also belongs to the community. This thesis is not the place to debate the merits of writers such as Derrida, since its aims are far more local. But it must acknowledge those ideas in the theory of writing, currently the scene of much speculation, especially as the recent ideas have much to offer AI.

One very important issue that seems intimately involved with the nature of AI is this relationship between speech and writing. Derrida has questioned the conventional distinction between the two and argues for a concept that contains both writing and speech. Taking Saussure at his word, he insists that if language is a system of differential signs, and meaning sits within the structures of relationship, and not in an ideal correspondence between sound and sense, then the definition of writing applies to every form of language.

To cover this wider interpretation he has even proposed a new term 'Arche-writing' (Norris 1987: 93).

In the same way that Derrida believes the sharp distinction between speech and writing is erroneous, in that writing seems to infect the pure spoken language of thought, so others have sought to locate aspects of thought within apparently dead writing. The role of written forms, either out there, or internalized into human thought processes, has become an active area of research. Thinking is shaped by writing and thoughts take on a new life when expressed in written form. This has provoked a new vision of the role of writing which sees it as more than 'speech writ down'. As Sokolowski summarizes:

> natural intelligence is not just an organic activity that occurs in a functioning brain; it is also embodied in the words that are written on paper, inscribed in clay, painted on a billboard (1988: 48).

Increasingly the traditional view, which sees representation as in the mind alone, is being modified by the recognition of the importance of the representations we use. In other words, "what it means to be humanly intelligent is tightly bound to the use of external representations in language and other media" (Hill 1989: 30). These representations arise to facilitate communication, but they seem in some way to become symbiotically connected to the mind. Writing, it is argued, intimately engages the mind, indeed some have argued that the very nature of self is a product of a literary process (Bolter 1991). Empirical evidence suggests that writing also encourages certain forms of thought, including that very rational thought which AI seeks to reproduce. This ability to analyse critically makes us different animals. Ong (1982) in fact goes so far as to suggest that all major advances in consciousness depend on technological transformations of the word. The importance of viewing the computer as an extension of these writing technologies is, therefore, clear.

> When we introduce our students to computers, we are inviting the restructuring of their consciousness in ways we cannot yet delineate clearly (Troll 1990: 119).

Not surprisingly, the role that computers play in the interface between human intelligence and writing has arisen as a concern for those using computers in their own professional writing. Barrett, amongst others, has recognized that both the computer and writing share certain features: offering a means for reviewing and developing ideas, and using a symbolic language to fix meanings: "Thus there is a family of relationships that unites thinking, writing, and computers" (1988: xvii). Just as this thesis argues that this relationship means that many of the ideas of writing can be applied to AI, we also see the language of AI being applied to literary writing: "like computer programs, written texts are an exteriorization of mind, and all writing must, in some sense, seek to pass a Turing Test..." (Paulson 1989: 300). It is with this vision of writing in mind that the argument for integrating AI and writing is undertaken.

An Emerging View: AI as Writing

There has been a noteworthy development in the AI literature over the past few years. Probably due to problems that have arisen within traditional AI, many workers have begun to question its foundations and seek alternatives, a classic example being Winograd and Flores (1990). Currently one popular alternative is the concept that AI systems can be seen as a medium of communication (e.g. Basden 1983; Whitaker & Östberg 1988). The idea was developed by Mark Stefik in a paper entitled 'The Next Knowledge Medium', in which he suggested that we abandon the conventional model of AI. The abstract of the paper is less a summary of its contents, than a manifesto for an alternative vision:

> The most widely understood goal of artificial intelligence is to understand and build intelligent, thinking machines. A perhaps larger opportunity is to understand and build an interactive knowledge medium (Stefik 1986: 34).

The idea of the computer as a knowledge medium, as a *surface* on which to express and exchange knowledge, offers great potential. It avoids some of the the thornier problems of AI, such as the possibility of creating a genuinely intelligent device (as we rely more on the natural intelligence of the people involved), and gives us a more empowering application, allowing such systems to act as effective communication tools between humans - with whom control still resides. It is this view that, since Stefik, has become widespread. It is significant that many of those proposing alternatives approaches are substantial figures in the AI community, such as Winograd and Clancey. Clancey, for example, has similarly proposed the machine as a mediator to replace the intelligent system as a surrogate.

> The new perspective is to keep in mind that people will continue to talk to each other. Think of the program and the computer as something to put on the side, next to people, which will mediate and facilitate their conversation (in Sandberg 1991: 8).

Negrotti takes the same position:

> The AI devices ... set up a sort of *piping of thought* that reproduces reasoning according to well-defined rules, not a kind of artificial mind (1987: 89).

Winograd too has argued that the notion of a 'symbol system', upon which most AI systems are built, is inherently linguistic and that "what we duplicate in our programs with their rules and propositions is really a form of verbal argument, not the workings of the mind" (1990: 185). Like Stefik he also believes that the utility of any technology, such as AI, need not depend upon accepting the original ideology of traditional AI, because that technology can be appropriated for other means, for:

> The computer, like writing, is fundamentally a communication medium - one that is unique to perform complex manipulations on the linguistic objects it stores and transmits (Winograd 1990: 185).

Therefore, he goes on to argue, we can *reinterpret* the techniques of AI from this new position, and use them for a new and possibly more productive goal. The re-interpretation of AI in this new light is something that is also undertaken within this thesis, which shows how the new approach not only aids new system design, but also gives a new understanding of current work in AI. Indeed, many of the actual working systems being developed may be better understood in these terms. There is then no need for apologies for the fact that these systems do not function as intelligent machines, but recognition that they function very well as means to communicate the system builders' knowledge to the system's users.

The broad view of the machine as a mediator between people has subsequently been further refined by others, who have suggested that the computer can act as a mediator between people in a similar fashion to writing: that is, AI systems are a form of writing. This view has been particularly advocated by Bolter (1991) and Sokolowski (1988). More metaphorically, Van Dyke (1987) has argued that we can use lessons from literacy can suggest new ways of teaching 'computer literacy'.

Bolter has summarized much of the argument that I want to pursue when he argues that "the strategy of artificial intelligence is always to disguise the computer as text and make of it instead a persona" (1991: 184), which is fair enough when one considers AI's original stated aim. But if we follow Stefik, and drop this aim, we can then recognize that just as "books do not replace human intelligence; they present human intelligence in another form ... Computers should be understood in the same way" (1991: 188). Likewise, Sokolowski has written that it would be more fruitful if

> instead of simply comparing computers and brains, we should also compare the 'reasoning' of the machine with the 'stating' of the sign, and examine storage and representation as they occur in the machine and in writing (1988: 51).

That is, we would do better by not asking how the computer is like a brain, but by asking how the output of the computer is like writing (1988: 54). It is taking up Sokolowski's challenge that constitutes one part of this thesis, making a detailed comparison between the results of AI and specific attributes of writing. We will go further, however, and ask not only what writing can tell us about AI, but what the computational nature of AI can add to writing. Writing not only helps us understand AI, but AI can help us look to the future of writing. For what computers offer is the next stage in the long-term process of the externalisation of human knowledge, in particular the externalisation of "human reasoning or knowledge processes" (Rommetveit 1988: 48). Where the computer differs from the book is in this concern with processes. It is an ideal medium through which to represent the dynamic nature of much reasoning. It is this that we should seek to exploit.

Others have also begun to explore similarities between the specific techniques of AI and aspects of writing. For example, Briggs (1985) has compared the knowledge representation work of Schank, and the work of the Sanskrit grammarians. Within AI much effort has been expended in trying to develop unambiguous representations of natural languages. But similarly, in addition to the creation of many works of literary value, Sanskrit has a long tradition of grammatical studies. Amongst these was a method for paraphrasing sentences that is almost identical to current work in AI. As Briggs notes, "AI has been reinventing a wheel millennia old" (1985: 32).

More extensive research into the development and impact of writing, similar to the approach taken by Briggs and documented below, suggests that the similarities with AI are more than superficial. From this position, knowledge based systems (that form of AI that uses symbols to represent knowledge) can be seen in a new light, in which their role is not to *simulate* human thought, but to *stimulate* it, and to act as a medium for thought. In the same way that written texts act as a medium for spoken words, and film acts as a way of communicating what can be seen, so the knowledge medium acts as a medium for what can be thought. However, where the approach described in this thesis differs from the traditional AI view is that it thereby makes no claim that this is thought itself, anymore than it might posit that text is voice. This is not intended to suggest that AI has no role in modelling reasoning, only that in building practical systems for people to use, it is not a requirement. Furthermore, the close relationship that appears to exist between writing and reasoning indicates the intriguing possibility that AI does in some way simulate reasoning because of its very role as a form of writing. If, as many argue, many of our ways of reasoning are internalizations of forms of writing, then extensions to writing that embody inferential capabilities do, in a significant way, model reasoning.

There is an additional complication regarding the relationship between writing and the approach to AI suggested below. In fact, written text plays a dual role in the Running Text system design. Features of writing assist in the general design of the new system; that is they provide a source of potential design features. But writing plays a second role in the system to be presented: a major issue that arose, and is discussed in detail in the chapter 3, is that writing needed to use an already known mode of communication — speech — as a 'crutch' during its development. Thus, within the system proposed in this thesis, writing has a secondary role as the carrier for the implemented 'modes of reasoning'. In other words, the author and reader are aided in their interpretation of the implemented AI components of the system through links between these and conventional, written, texts.

But we now come to a paradox. Due to AI's history and nature (as essentially related to dialogue), AI techniques usually operate in the form of question and answer methods, where the machine is seen as an agent in the dialogue. Generally the user inputs a question and obtains the answer. How can a technology steeped in this tradition be employed as an expressive medium, allowing the authorship of various lines of thought? The solution employed in the Running Text system is to have authors ask *themselves* questions, a solution with a long history. They create representations of their view of topics and ask the relevant questions. The answers that the 'mode of reasoning' produces, along with the partial AI model required to generate it, form part of the final *text*. Many of these question-answer pairs are created and assembled in an (essentially) linear fashion to create the final document. These answers are then given context, and connected together, through their own links to a conventional linear text, which expresses the answers in a normal textual form and provides standard writing techniques to bring the whole 'document' together.

Used in this manner, the techniques of AI function as a form of 'alphabet' or 'vocabulary'. A final concern is to ensure that the vocabulary is large enough to be sufficiently expressive, but not so large as to be impossible to learn. Fortunately artificial intelligence has developed a variety of implementable techniques that can represent many of the various ways in which we think. Examples of such modes of thought are: causal, diagnostic and metaphorical reasoning. The various program types that implement 'modes of reasoning' can be seen as our putative cognitive alphabet or vocabulary. An author can create a new form of text by assembling a series of such modes of reasoning to construct an argument.

A consequence of our notion of AI as writing, and implicit in the new ideas about writing, is a new approach the nature of representation, be it in minds, machines or texts. This is far too large an issue to cover effectively in a thesis that has other topics as its main concern, but one or two particular arguments are worth pointing out as relevant. Traditionally, it has been the "basic premise of AI that knowledge of something is the ability to form a mental model that accurately represents the thing as well as the actions that can be performed by it and on it" (Sowa 1983: 2). However, another approach is that used by Hacking (1983: 133). For him representations are, by their very nature, "external and public". These are two characteristics which will constantly arise in our discussion of AI, for, it is suggested, AI programs share these characteristics. In the same vein, Clancey has argued for a 'Copernican shift' in our view of memory, and for us to "put our representations out where they can be perceived, say as silent speech and visualizations, but mostly in our writing and in our drawing" (Clancey, in Sandberg 1991: 5). We seem able to reason not only in the mind, but through these external means, and latterly through internalized forms of these external representations. The separation between world and model, inherent in the traditional view begins to break down.

What brings the qualities of being external and public together is that they are features required for communication. To communicate is "to turn one's thoughts and feelings into common formulas" (Larsen n.d.: 333). The etymology of the term 'communicate' is 'common', which means 'pertaining or belong to the whole community'. Therefore communication is a process of formation, and its product is 'information'. For Larsen there is a fundamental difference between information and knowledge, which is associated with the process of communication: when personal knowledge is communicated it becomes public information. A similar point has been made by Walter Ong who argues that knowledge is not like an object which can be transferred from one person to another:

> I can only perform actions - produce words - which enable you to generate the knowledge in yourself...Since knowledge cannot be physically transferred verbally from one human person to another but must always be created by the hearer or reader within his or her own consciousness, interpretation is always in play when one listens or when one reads (1986: 38).

Following this argument would suggest that Stefik's 'knowledge medium' may be somewhat misnamed. If the computer is being employed as a means of communication then we have information, although it is of a significantly different nature from the static data with which the term of often associated. Its identification with knowledge comes with the 'computer as agent' metaphor which is currently under question. (Despite these concerns, I will continue to use the phrase 'knowledge medium', due to the significance of Stefik's paper in the literature, and his introduction of the term). However, if we seek to use the technology to act as a medium for communication between people the characteristics of symbolic AI representations, that of being external and public, are invaluable. In fact, the need for communication is crucial to the argument presented, it being a reason for making our thoughts external and public (in a machine) in the first place. The issue of our machine representations being public, as a pre-requisite for its use as a medium, also raises questions about the role of non-symbolic, connectionist AI systems in the 'AI as writing' approach. These systems have their representations buried within the network and thus not visible to the user. For this reason, the term AI in our arguments is strictly to be interpreted as referring to symbolic AI, and the implementations described in the following chapter of this form.

2. AI, Dialogue and The Computational Tablet

One of the important themes within this thesis is the perception that we, as human users, have of the computing systems that result from AI research. What are our feelings towards the machine that we are using? One very useful classification (Hill 1989: 35) describes our perceptions as first-, second-, and third-person experiences:

- A first-person experience — is one where users feels that they are solely doing the intellectual work, they sense themselves, their *I*.
- A second-person experience — is one where the user sees the program as an other, a *you*, an agent to converse with. And finally,
- A third-person experience — is one where the user feels the presence of an author, a *he* or *she*; the program is an artifact created by another person.

Recent work in traditional, tool-building, computing takes an essentially first-person approach; in using a word-processor to write these words, I am aware of myself and what I am trying to write. Traditional AI takes a second-person approach, one is expected to be aware of the machine as another self. An example of a third-person experience would be using an e-mail program, where one is primarily aware of the author of the message, of another person. In these terms, what the work described below aims to do is to take AI systems, and make of them a third-person experience — a medium. As our main concern is with alternatives to traditional AI, some time needs to be devoted to a discussion of the image of the machine as an other, in order to place the new approach into context.

The Turing Test

One of the most influential elements in discussing the history and goals of artificial intelligence is the so-called *Turing Test*, a test which aims to be able to answer the question of whether a computer is intelligent. As Halpern (1990) notes, the paper in which the Test was introduced has been reprinted many times, is cited frequently in the AI literature, and has even been the inspiration for a television play and a novel. It is a paper and an idea whose strange career Halpern has critically investigated. What is important is the way that in which it has moved into myth, and Turing's actual test is often forgotten. Exactly what was Turing setting out in the Test, as described in the original paper?

The test is based on, and an extension of, what Turing calls the imitation game. This is played by three people: A man, a woman, and an interrogator, who can be of either gender. The interrogator is in a room separate from the other two, but can communicate with them by a teleprinter. The aim of the game is for the interrogator to be able to assign to the two others their correct gender. The use of the teleprinter is to ensure that the interrogator cannot use voice or appearance to help, but must make deductions only on the basis of responses to questions. Turing then goes on to ask:

> What will happen when a machine takes the part of [the man] in this game? Will the interrogator decide wrongly as often when the game is played like this as he does when the game is played between a man and a woman? These questions replace our original, 'Can machines think?' (Turing, in Boden 1990: 41).

However, popular works on AI often describe to the Test in the following terms:

> if the interrogator could not tell for certain whether he or she was communicating with a person or a machine, the machine could indeed be said to think (Feigenbaum & McCorduck 1984: 156).

But is this really a reasonable extension of Turing's original argument? In the case where the interrogator has to determine that gender of the subject, if the man manages to persuade the interrogator to assert that he is a woman, does that make him one? Is he now a woman rather than a man? Surely not. The test may say something about our prejudices about masculinity and femininity, in terms of expected behaviour, but the essentially biological facts are not changed. Thus, the assertion that a computer which convinces us that it is a person has intelligence is weak. Under that criterion, the written message would equally pass as intelligent, as in the cases described later in chapter 3, where members of non-literate cultures thought that written messages could speak and observe.

Sokolowski (1988) draws a relevant comparison between various uses of the term artificial. Thus artificial light, created by man-made means, is still light, the same as light from the sun. However, an artificial flower, made of paper, is not the same as a real flower, but only appears to be when seen under bad lighting conditions and from certain angles. The question of course is whether artificial intelligence is of the former or latter type. For Halpern (1990) it is more of the latter. That is, it may succeed in fooling us for some time, due to the restricted nature of the test, but eventually, in the right conditions we will identify it as not being a genuine intelligence. Just as a man can only pretend to be woman so far — eventually (and how far is a personal point) an interrogator will identify the subject as male. One could push the argument into situations of indecision, but these are ones where we finally begin to question our notions of gender. The Turing Test then, which tried to avoid dealing with issues of the real nature of intelligence (Turing's intention in devising the test was to avoid having to deal with definitional questions such as 'What is intelligence?'), merely delays them. Rather critically Halpern thinks that Turing is re-stating the question 'Can a machine think?', as 'Can a computer be programmed to fool a human being into thinking that it is another human being?" (1990: 7).

But the significance of the Turing Test in the development of AI comes as much from its procedure as from the idea underlying it. The use of the term *interrogator* makes the point that the test occurs in a situation of asking questions and receiving answers. There are reasons for understanding why Turing and others might see this as a means to uncover intelligence, but, I believe, this has less to do with the imitation game and more to do with the logocentrism that Derrida identifies; a dialogue implies conversation, which further implies voice and hence self-presence. To test children on their intelligence, it is usually the case that they write extended essays on a subject, which is assessed. Would it not be as reasonable to devise a Turing type test in which the machine has to write a long text explaining a topic of some complexity? Somehow the interaction of dialogue seems to indicate a presence, which the evidence in an essay does not. This may also explain the use of oral examination in those cases where the examiners want to be more certain of a candidates true understanding, such as borderline cases or higher degrees.

Expert Systems and The Role of the User

The tendency of users to attribute agency to the machine may be a measure of success to those working in AI, such systems could possibly pass the Turing Test for some time, but it also brings problems in the real world. One question that arose earlier is why should AI look to alternatives to the 'machine as agent' metaphor, and thereby possibly abandon its original goals. There are, of course, a number of doubts surrounding the possibility of the creation of intelligent, non-human, systems. The goal of creating a knowledge medium rather than an intelligent system need not intend to replace conventional AI, as Stefik desires, but can be a worthwhile goal in its own right. However, the rationale for undertaking the effort required to develop a knowledge medium can be found by looking at the debate that surrounds these problems and usability of those AI systems that have been built, for the sense of an 'other' moves the feeling of control away from the human user and towards the machine.

Most work has been on users' reactions to the most prominent of the AI technologies, expert systems, as they are the form of AI technology that has had the greatest success in terms of real-world applications. Expert systems usually encode an expert's domain knowledge of a particular area as a set of if-then rules. This knowledge is in a limited and often technical domain where one would call upon an expert, rather than everyday common-sense knowledge, hence the term 'expert system'. The user comes to the system hoping to have a specific question answered. A typical example would be a diagnosis of a patient's condition or the fault in a piece of equipment. The system will engage in a dialogue with the user, eliciting the facts already known about the problem, possibly requesting that the user go and obtain further data, and finally, on the basis of following the if-then rules, come to a conclusion which is given to the user. The user is then in a position to act upon the advice given.

Various problems arise with this form of interaction. The dialogue with the machine gives the user a sense that the machine is 'intelligent', and the user may, not unreasonably, expect the system to have a wide range of experience and understanding, like a human expert. Mostly though, the machine has functions in only a very limited domain, and questions outside of that domain will lead to meaningless results. This characteristic of expert systems is known as 'brittleness'. One solution to the problem of brittleness is to be found in the work of Lenat, Prakash & Shepherd (1986), which is very much within traditional AI. They believe that an AI system must contain a wide range of common-sense knowledge, enabling it to cope with 'unexpected' situations, and are creating a such a large knowledge-base (called CYC) of real world facts. This is a significant requirement to be met, both theoretically and practically. The alternative is a product of the new metaphor. Nobody expects a book to be able to answer all the problems one might have, skimming through it, seeing its size and the table of contents, gives the reader a pretty good idea of the breadth and depth of coverage. If an AI systems could project a similar form of humility, the problem of brittleness may not be solved, but it will not induce unfavourable reactions in its users.

Even when they function well, that is, give the correct answers, there are still many problems with user acceptance. It is argued (Woods 1986) that, in effect, they reduce the user to a data input, and solution implementation, peripheral. In other words, the human user must answer the questions that the machine asks in its requirement to accumulate the necessary information. And finally, the user is supposed to filter-out (that is, ignore) any poor solutions, assuming he or she is still capable of doing so, and implement the suggested solution in the real world. Interface design thus becomes an issue of using the human user as an interface between the machine and the world, rather than having the machine as an interface between the intentions of the system designer and the system user. The machine cannot engage in the real world and relies on the human as a proxy to do so. The application of AI to medicine, up to now considered one of its most successful areas, has found difficulties such as these with the traditional 'consultation' role — and this is despite evidence that the answers the systems give are usually very good. For example, it was demonstrated experimentally that MYCIN (the archetypal medical expert system) performed diagnosis at the level of human expert doctors, but the system was eventually abandoned and was never employed in a practical medical setting (Lipscombe 1989).

These systems were built on the metaphor of the machine as an expert partner, as embodied in the name 'expert system'. But the users were just not happy with relinquishing their decision-making to a computer. Not only is there a machine partner who has some control over the interaction, but it is supposed to be an expert partner which inevitably reduces the user's sense of control further. The expert system's role as a computer expert to the user novice was one that did not appeal to the majority of users who felt threatened in some way. One particular problem that Woods (1986) identifies is the issue of responsibility and authority. A computer consultant (expert system) provides a solution to specific problems, but the human user usually has responsibility for the outcome of any actual decision — only in more speculative texts does one find discussion of the need for AI systems to have legal responsibility. Naturally, the user can filter the computer's solutions, but such an interaction has certain side-effects. Woods finds two reactions to what he calls this 'responsibility/authority double-bind'. Either the user rejects all the system's output, finding or creating grounds for doing so; or alternatively, he or she abrogates decision-making to it, that is, the user never over-rides the computer if the cost of overriding is considered too high. Problems where the machine comes to dominate the human user then arise, for example:

> an experienced integrated circuit designer is forced to make an unfortunate choice: Let the machine do all the work or do all the work himself. If he lets the machine do it, the machine will tell him to keep out of things, that it is doing the whole job. But when the machine ends up with five wires undone, the engineer is supposed to fix it. He does not know why the program placed what it did or why the remainder could not be handled. He must rethink the entire problem from the beginning (Feingold, cited in Woods 1986).

If we expect that the user should be able to filter properly, then one is basically asking that the user be as proficient in the task as the machine, which now acts, in effect, as a form of check-list of crucial issues, rather than an advisor. The need for competent users also gives a lie to those who try to sell expert systems as a means to employ unskilled and cheaper staff. As soon as the 'expert system' comes across a novel situation, the advice will be inappropriate, and any savings accrued can be wiped out in one serious error.

We see a process of designers moving away from the idea of agency, for example, in the program ONCOCIN, which is a direct successor to MYCIN (Lipscombe 1989). The developers of the ONCOCIN program intended that the system should actually be used in the field and so abandoned many of the 'strong AI' demands (the approach to AI which sees it as a simulation of human cognitive performance), it acts less as a agent. But another means they employed to encourage use was to ensure that it *had* to be used. The doctor is forced to take the patient protocol data from the machine and to enter the data for the current patient visit into the terminal. The program automatically fills in recommendations for therapy and test. The designers claim that the doctor is still in control as he or she can override ONCOCIN's decision. But as Lipscombe observes "[a]lthough Shortliffe [the program's designer] talks about the doctor being in control, ... the doctor has ultimate responsibility" (1989: 191). Moreover, to be able to evaluate the programs recommendations, in order to decide whether to ignore them or not, the doctor has to do what he or she would normally do, that is, solve the problem. Sørgaard identifies the same problem. The performance of an expert system will typically degrade drastically at the edge of the system's area of application, therefore "the person using the system must have the competence and common sense to judge whether the advice given is reasonable" (1991: 11).

Problems of this kind are unlikely to encourage the use of expert systems. The suspicion is even more understandable when potential users of expert systems read that:

> Other benefits are that the computerised expert or advisor is always alert, is never under the weather or temperamental, can work 24 hours a day, seven days a week, and cannot accept a better offer from a business competitor (Manuel 1986: 60).

In the same spirit, someone actually building AI systems, Robert Milne of Intelligent Applications, clearly states that he believes the business benefits of expert systems come "by reducing the skill needed for a particular task, the company is able to reduce training needs, [and] use lower pay rates for employees" (1990: 2). To help managers in the task of deskilling, Allen (1986) even provides advice in estimating the costs of an expert to an organization, to evaluate the potential for building an expert system to replace him or her. But as we have seen, this is a potentially dangerous path for any business to take. Less dramatically, but more plausibly, both Berman (1992) and Östberg (1988) view knowledge engineering as a contemporary form of 'work study engineering', one that divides up the cognitive tasks much as manual tasks were divided up, leaving information workers with unfulfilling work.

Explanation and Critiquing

The classic solution to the problem of authority, and the need for users to be able to defend their actions, is the provision of an explanation system as part of the expert system. Instead of just giving the straight answer to the query, an explanation is generated which usually consists of a trace of the line of reasoning that lead to the final conclusion. The user is then in a position to identify any inferential steps with which they disagree, and possibly change them. One important point of such systems is that once the user can see the knowledge base (the set of rules that leads from the facts to the conclusion) and edit those parts of the system with which they disagree, then we have left the notion of the 'expert system as agent' behind. After all, we do not have such access to the minds of other people. But we do have access to see, and to edit, the written words of other people. Although explanation is promoted as an extension of dialogue, its provision can be seen as starting to move us away from the conversational mode and towards the idea of the system as a 'writer of essays' and not a participant in *viva voce*.

In fact, the characteristic of the explanatory component to give users access to the workings of the system has been an aspect that has discouraged its general use. Woods (1986: 89) has seen developers of expert systems show serious concern about providing explanations, as they constitute enough information for competitors in the expert system business to build a comparable and competitive system. Similarly, the companies that have bought these systems for their staff dismiss explanation components as they may increase the skill of the user, who could then leave and join a competitor.

The potential of explanation, that is a fuller description of how a result was obtained, to communicate information and satisfy users has been documented by Coombs and Alty (1984). In a study of a university computer advisory service (manned by human advisors), student users, although given an answer within 10 minutes (and, for the most part, a correct answer), found the advice sessions unsatisfactory. The style of the interaction was similar to current expert systems, in that the expert asked questions to determine the relevant fact of the problem, and then stated the answer to the problem. What was missing was feedback, elaboration and explanation. It is the difference between having an answer and having some understanding of the problem. Katona (Simon 1986) showed in a series of psychological experiments that when an explanation is provided as to why a given solution works, the solution is better remembered and is better transferred to similar but non-identical problem situations, than when no explanation is given. A knowledge medium would aim to exploit this by making the communication of knowledge the primary goal, rather than an profit-threatening side-effect.

Another solution to the non-acceptance of consultant expert system has been 'critiquing systems'. In this approach the system does not act as a consultant but critiques proposals made by the users, pushing the centre of gravity in decision-making towards them (Langlotz & Shortliffe 1989). The first explicitly 'critiquing' program was ATTENDING, which was used to critique an anaesthetic plan. The plans (which are devised preoperatively) are critiqued by evaluating the risks associated with certain anaesthetic procedures, given characteristics of the particular patient. The user's proposed plan is critiqued by taking its associated risks and comparing them with those of alternative plans. This are an obvious improvement over the straight expert system, but critiquing systems have also been shown to suffer from user acceptance problems. For example, the JANUS system (Fischer, McCall & Mørch 1989) uses knowledge-based critics (agents that critique) to 'look over the shoulder' of the designer during use and critiques certain decisions. However, users only received the information when they had made a mistake, which they found very frustrating (McCall et al. 1990). It would probably have been better if they could have got a feel of the issues involved in making good decisions, prior to doing so, rather than making many simple design errors.

The original intention of both these extensions to expert systems, the explanation and the critiquing, are, however, still embedded with the 'machine as agent' metaphor. What has been done has been to try to change the psychological profile of the simulated partner so as to assist the users more, and annoy them less. Others have gone further and looked away from this metaphor entirely, although without specifically seeking another. An example of this more contemporary approach is that based on the work that was originally seen as being a successful expert system: INTERNIST. The developers have abandoned their original intentions, despite early evaluation successes (in terms of giving the correct answers), and have given the system a new name to accompany the new philosophy. Now called QMR (Quick Medical Reference), the system no longer attempts to act as a high-level advisor, but operates more as an information management tool. It provides users with multiple ways of viewing and modifying the diagnostic information. For example, should you want to know how many alternate ways that a patient with a cough could also have kidney disease, QMR will provide, in a few seconds, a hundred ways in which this is possible, ranked in order of likelihood. It does not act as a partner (be it expert, apprentice or critic) in a any way. A better way of seeing the system behaviour is as a complex index, getting the user to a set of entries matching certain requirements. It is now so far from AI, that one would have doubts about applying the term at all, although it has exploited the techniques of the original INTERNIST system to achieve these results.

Similarly, the PUFF expert system is used as an aid. It is intended that the user be able to quickly recognize, and hence ignore, any incorrect advice. What it offers the user is savings in time over their usual methods

> because the advice already is (sic) nicely printed ready to be forwarded as the expert's own conclusion. The cases where the system's advice is faulty or incomplete can be handled in a more conventional way (Sørgaard 1991: 9).

What is particularly interesting about this description is that one of the most valuable aspects of the system was the printed output. As will been discussed elsewhere, what separates oral from written culture is the ability to be able to review and pass onto others the results of reasoning. The traditional expert system, being based on the image of a dialogue, seems to promote a similar transience of results. Once the dialogue with the expert system is complete, the user goes away and implements the answer and the reasoning line is lost. The PUFF example is better seen as a more complex form of writing aid, citing the work of others — those others being the system's original designers. If a manager comes away from a computer system having used it to write a report which may include a spreadsheet and e-mails from others, we still have no problem in ascribing to the machine the role of an aid to the authorship of a document. We have no reason to argue that the machine has authored the document, although in a very limited sense it certainly 'created' the print-out which sits before us.

The lesson from these developments is that users appear not to want to have a clever partner. That goal seems to come equally from researchers who see interesting problems to solve, and from system vendors who often believe that users want simple answers and a simple, conversational, interface. But as the success of the desktop/windows interface shows, the solution need not be conversational. However, once the 'computer as agent' metaphor has been abandoned, as it increasingly is, there is a conceptual void which needs to be filled. A new and coherent approach is required to assist in the design and evaluation of new systems. The notion of the computer as a surface to write on and read from provides us with a viable and useful framework.

Reconstructing Expert Systems

If expert systems take a second-person role, and this is undesirable, can we modify the design of these systems to be otherwise? That is, given the same set of data and rules encoded in the system, is there an alternative means to access this material, other than through dialogue. We can see by a rewriting of existing system interface how alternative approaches might appear. Take, for instance, a sample 'conversation' with Prospector, one of the most prominent expert systems ever constructed. Human responses start with a double asterisk (Kurzweil 1990: 300), otherwise they are from the machine.

I am considering the possibility of a (Type-A porphyry copper deposit - PCDA) in the target area. The following questions are intended to establish the nature of the petrotectonic setting. 1- To what degree do you believe that: (the target area is in a continental margin mobile belt)? ** -2 The following questions are intended to determine the nature of the regional environment. 2- To what degree do you believe that: (there are granitic intrusives in the region)? ** 5 3- To what degree do you believe that: (the region contains an abundance of small stocks)? ** -3 ... 8- To what degree do you believe that: (igneous rocks in the region have porphyritic texture)? ** Why I am trying to establish whether some of the intrusive rocks in your area have textures suggestive of a hyperbassal to subvolcanic environment. Porphyry textures resulting from the rapid cooling and crystallization of a silicate melt which contains large mineral grains are suggestive of relatively shallow levels of emplacement. 8- To what degree do you believe that: (igneous rocks in the region have porphyritic texture)? ** 5 My certainty in (the texture and morphology of the intrusive system suggest a subvolcanic (hypabyssal) regional environment) is now: 4.924. Do you want to pursue this further? ** No

Fig. 2.1 Conversation with Prospector

There are a number of features of this interaction that encourage the user to attribute 'agency' to the program. Firstly, the interaction consists of a series of short exchanges: a dialogue. Conventionally a dialogue is an exchange that we have with other people. Secondly, the program is written in such a manner that the personal pronouns 'I' and 'you' are used to refer to the program and user respectively. This strategy is one that we also see in the early history of writing. Greek soldiers in Egypt cut the following 'graffiti' on the legs of Rameses II's rock-cut colossi at Abu Simbel:

> When King Psamatichos came to Elephantine, this they wrote, who sailed with Psamatichos son of Theokles. They went beyond Kerkis, as far as the river let them; Potasimto led the foreigners and Amasis the Egyptians. And Archon, son of Amoibichos, and Peleqos, son of Eudamos, wrote us (in Burn 1983: 95-6).

In this quote the final 'us' refers to the letters themselves. That is, "the stone speaks, as in many early inscriptions" (Burn 1983: 96). We would now view this as a literary ploy, and, it can argued, so too is the use of personal pronouns by AI software.

This clearly conversational interaction can be modified to function as an active document processor. In the following reconstruction, the author has more control, and the system acts as a complex text-processor which is more active in encouraging the production of text. Essentially, one need only remove the aspects of the interaction where the machine is using 'I' to refer to its own processes and 'you' in its exchange with the user. Furthermore, the explicit exchange of the dialogue, which implies a partner, can also be altered. One can imagine that it might actually work thus: The author engages a form of word-processor (WP) which has all the features found in a typical WP package. In addition functionality from the expert system is integrated; thus by pressing a mouse button at the insertion point for the developing text, brings up a menu of options for both data input, and other typical expert systems functions (solve and explain). The new data, solution or explanation is added to the text, at the insertion point, by the system, much as a computerized thesaurus inserts new words at the will of the author. In the mockup below the user's direct text input is marked by being in italic and the text that the user elicits from interaction with the computer is in plain type.

Report on survey. Author: John Smith...

The next issue we need to investigate is the possibility of a (Type-A porphyry copper deposit - PCDA) in the target area. To establish the nature of the petrotectonic setting:

- degree that: (the target area is in a continental margin mobile belt)?*is I believe about -2*

To determine the nature of the regional environment:

degree that: (there are granitic intrusives in the region)?*is about 5, due to the results that David Warwick recorded last week*

degree that: (the region contains an abundance of small stocks)? *is 3, given the data in report number TR-78.*

...

8- degree that: (igneous rocks in the region have porphyritic texture)?

** *Why is this important?*

because I am trying to establish whether some of the intrusive rocks in the area have textures suggestive of a hyperbassal to subvolcanic environment. Porphyry textures resulting from the rapid cooling and crystallization of a silicate melt which contains large mineral grains are suggestive of relatively shallow levels of emplacement.

degree that: (igneous rocks in the region have porphyritic texture)?*is about 5*

Therefore, the certainty in (the texture and morphology of the intrusive system suggest a subvolcanic (hypabyssal) regional environment) is now: 4.924.

Thus I suggest the following actions...

Fig. 2.2 Reconstruction of Prospector as a Writing Aid

Used in such a fashion, the resources of the expert system are more like a writing tool, such as a spelling checker or thesaurus. It helps us refer to standard known forms of expression in certain areas of knowledge, to add to what else we want to communicate in normal text. It structures the document in the same way as a template for other specific document types might. For example, authors writing a scientific paper are expected to use certain conventions: abstract, introduction, methods, results, conclusions and references. Equally, most e-mail systems request that the user place their information into a template, giving the addressee, the subject, and then the message; the system adding information on the senders address and the date. Expert systems used in the mode described above can be seen in a similar light, but operating at a finer grain, down to the sentence or clause level. This issue will be discussed in detail in chapter 5, when we investigate structured hypertexts.

A 'Word Expert'

If the argument is that the choice of interaction style between the system and user is a matter of choice, one could equally convert a tool-based system to a agent-based system. For example, one could, if so desired, rewrite such tools as a spelling-checker or thesaurus, to be a 'word expert' system, and to operate in a dialogue manner, but few would argue that any benefits would accrue. Let us make a comparison.

Within Word, the word-processing system that is being used to write this thesis, there is a spellchecker which looks like the figure below when it comes across the word 'conventionalize' spelt wrongly as 'conventinalize':

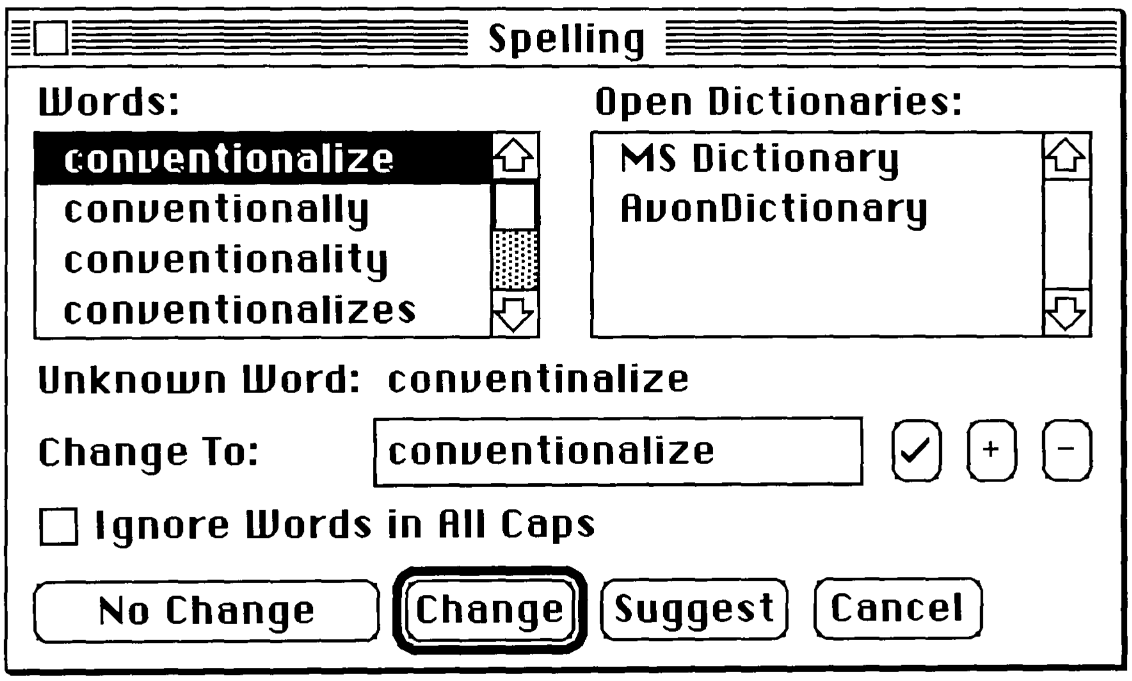


Fig. 2.3 The Word SpellChecker Display for 'conventinalize'

If it finds an unknown word, the user can view alternatives (based on a simple matching algorithm — nothing very clever here, but quite useful), which appear in the left-hand list pane as a set of alternatives. The user can then select one and have the misspelt word replaced, or have a new word added to the dictionary if the checker has simply failed to recognize a word the user wants (such as a technical word or personal name). The thesaurus, called Word Finder, for the same word has the following appearance:

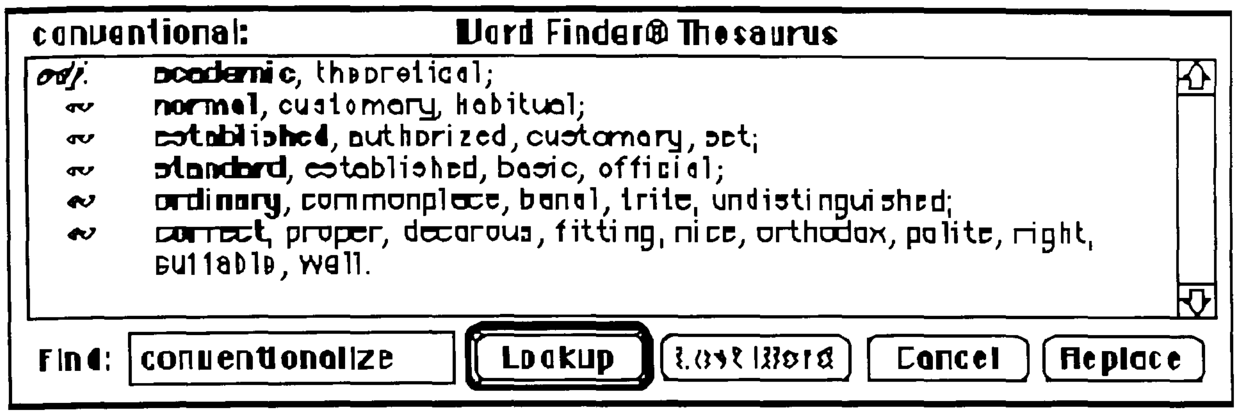


Fig. 2.4 A WordFinder Display for 'conventionalize'

It operates in a similar fashion to the spelling checker. Various alternative words, distinguished by their broad interpretation are listed, and the user can, using the mouse, select one and have it replace the original word in the document being checked.

This kind of operation is, by Hill's categorization, a first-person interaction. The user is only aware of what he or she is trying to say. But it also not difficult to imagine an expert system in the domain of spelling and word use, which one could also implement. An interaction with a 'word-expert' might appear thus (with acknowledgements to Prospector), if the normal expert system paradigm was applied.

What word are you interested in?

** conventinalize.

I do not recognize this. Do you mean (conventionalize)?

** Yes

Do you want to see any similar terms?

** Yes

I do not have any terms for (conventionalize), would (conventional) suffice?

** Yes

Do you intend (conventional) to be understood as: (academic, normal, established, standard, ordinary, correct)?

** established

Other terms are: (authorized, customary, set)

Fig. 2.5 Conversation with a 'Word Expert'

The question is whether we gain anything from implementing such an interface, other than trying to convince the user that there is an intelligent agent sitting within the machine. The conversion from the existing design, a tool-based one, to the expert system design would be relatively trivial, and the latter could be easily implemented. Again, if the technology available to the system designer was a glass teletype, it is understandable that this form of interaction would spring to mind. The graphic capabilities of current machines, however, allows us to adopt the actual Word style.

Transparency and Modifiability

A simple first-person interface to expert systems alone is insufficient for the purposes of this thesis, creating not only a writing tool, but a medium for communicating knowledge. The modification of the Prospector expert system above, retains some of the problems that are inherent in expert systems. If we look again at the modified input, it has the appearance of a tool, a first-person implementation. One difficulty is the formal appearance of the output which may be unacceptable to readers, until such time as it becomes conventional and understood by users. There is, however, a more significant issue, that of transparency and modifiability. For, as Hill (1989) has pointed out, our representations are commitments to particular ways of thinking about the world and thus should be able to be challenged. He argues that it is, therefore, important that any representational media created by AI researchers should empower users by enabling the user to modify relatively easily these so-called facts. The kind of interaction proposed above for the Prospector systems does not facilitate such modification. The essentially authored role of the original and underlying knowledge-base is not made explicit. Thus, it would be an improvement if the authored nature of the rules within an expert system could be made clear for the user, they should be accessible. Furthermore, it would also be of benefit if users had the ability to edit these rules and facts if they disagree with them. These considerations point to other ways of implementing an AI system that makes the third-person view of the system clearer. It is one design criteria that motivates the Running Text in Chapter 6.

The issue that arises from all of the discussion above is that any knowledge medium can, and probably should, avoid the 'machine as agent' metaphor. Interestingly, the concern about the questionable effects of the machine using the 'I' and 'you' may be only a temporary one. Personally speaking, I have found that having now spent some years actually working on the idea of the machine as a medium of others' thoughts, I can now view a conventional expert system and can easily see it as not being an agent. Indeed, I have actually to suspend belief for a while to see it being any form of intelligent, autonomous agent. A parallel case exists in ordinary writing. When you, the reader, see the term 'I' in this document, you associate it with me, the author, not the document itself.

What is crucial to the argument is that the machine can be seen as a new writing surface, one that has *dynamic* qualities that *structure* the material being written upon it. It is this structuring that is critical, aiding the thoughts of the author and guiding those of the reader. As with many other tools, such as the nomogram, it is the structuring of the cognitive environment that enables work to proceed smoothly. In the early days of writing, people found that the act of writing on a tablet encouraged a new way of understanding the material written, hence constituting an advance over the spoken form. Similarly, the reconstructed expert systems, in a form that structures the writing being made on it could be called a *Computational Tablet*.

3. Writing and Artificial Intelligence: Parallels and Themes

This chapter explores the relationships between AI and the earlier technologies of the word: writing and print. It will show that these technologies and AI share a number of features which suggest that they can be seen as instances of the same class — writing in a broader sense (but not as broad as Derrida's arche-writing). This will involve a general introduction to the nature and history of writing to establish a context, followed by a more detailed discussion of those features which writing shares with AI, or which in some way relate to developments exhibited by AI.

The parallels between AI and writing serve three roles:

- They provide evidence to support the claim that AI can be seen as a form of writing.
- The solutions for various problems found within the history of written and printed text can inform the design of a possible knowledge medium, the Running Text system
- The impacts of writing and print indicate possible impacts for AI used as a medium.

At the outset writing, and AI's parent computing, share one obvious feature. It has become almost a commonplace that computers are indispensable in our society. Even though it is a recent human invention, it is difficult to imagine how we could continue without the information technology that now surrounds us. In both its importance and its artificiality, it is matched by writing. Despite its (historically) recent development, it is also difficult to imagine our world without writing: no newspapers or books, tax documents, or road signs. Most religions would be vastly different, as would our system of law. Modern science would be almost inconceivable. But unlike speech, writing is an invention. It appears that we are born to speak; there are no known societies which lack it. On the other hand, relatively few of the languages ever spoken have been written down. It is only the success of the cultures which possess writing that blinds us to this fact. It is the nature of writing which has put it in such a strong position, to which we turn.

The Nature of Writing

Prior to any discussion of the writing, we are faced with the immediate problem of what we are going to include within our definition of writing. For one of the major issues that constantly arise in any study of writing is that the term is employed for a number of purposes. It has already been noted that the 'AI as of writing' metaphor can be interpreted in two ways. Either we can employ it as a straight metaphor, where similarities can be found and other features used to reassess AI; or we can broaden the use of the term 'writing' so that it includes not only the kind of writing we all do every day, but also other forms of expression and communication. The argument for considering the second case has recently become particularly relevant, as post-structuralists, for their own reasons, have sought to appropriate the term and apply it in a broader sense than that conventionally understood. The deconstructionists argue that:

> writing is not displaced speech but a primary act. This argument defines" writing" in a much broader sense than we are used to, to mean something like "making public" in any manner, including speech (Bruffee 1984: 641).

So we have two ways of viewing writing. The older and more intuitive is to see it as contrastive with speech, the foundation of the whole orality versus literacy debate. Another, more contemporary (but more controversial) view sees both speech and conventional writing as forms of a broader writing. The latter view makes sense when applied to a discussion of the psychological and social impacts of writing. It has been argued that thought is not an 'essential attribute' of the mind, but that thought and conversation are causally related. In other words "thought is an artifact created by social interaction" through speech (Bruffee 1984: 640). If our thinking has its roots in conversation, then any new tools that extend and modify that conversation, as writing technologies do, must impact on thought itself. Given this view of writing, the relationship between it and artificial intelligence is more than a set of potentially interesting parallels; they are intimately connected. However, let us begin by looking at the narrower definition of writing, providing firm ground from which to start, and move onto the broader definition when appropriate.

Writing and AI: some similarities

Coulmas (1989: 17) lists the following as fundamental characteristics of writing:

1. It consists of artificial graphical marks on a durable surface;
2. Its purpose is to communicate something;
3. This purpose is achieved by virtue of the marks' conventional relation to language.

Similarly, for Gelb, writing is "a system of human intercommunication by means of conventional visible marks" (1963: 12). These definitions are very unspecific, and indicate how widely 'writing' could be interpreted. It would be a simple matter to create a computer system that exhibited these characteristics, indeed it would be difficult to find many existing systems that do not. Therefore, this chapter will take another strategy and look into the specific developments of actual forms of writing, and other similar notations, to argue the case that AI shares their features.

The first parallel is one that strikes immediately as indicating that writing and AI share common interests. One of the most curious facts (from our contemporary perspective) is that written texts, like AI systems, were often considered by nonliterate cultures as 'autonomous and intelligent'. Gelb (1963: 232) has remarked that for certain cultures, "a book is considered as a living being which can *speak*". He recounts (1963) a number of interesting stories on this theme. According to one, a native messenger refused to carry a message because he was afraid that the letter might talk to him while he was carrying it. In a similar example the messenger would only carry the message after he had pierced it with his lance to prevent it from talking. There is also the tale of an Indian who was sent to a missionary with four loaves of bread and an accompanying letter, upon which this number was written. He ate one loaf and was found out. When asked to undertake a similar errand, he repeated the theft, but took the precaution of hiding the letter under a stone so that it could not see him commit the second crime. Another comparable story from Australia reports a native who was carrying a package with an accompanying letter. He stole some of the tobacco from the package and was astonished when the recipient was able to find him out, although he had hidden the letter in a tree — in his anger he subsequently beat the letter furiously. The Cherokee people used the phrase 'talking leaves' to describe the leaves of paper that have no voice and yet can be made to speak (Harris

1986). This phrase indicates how one might also understand AI systems. The system has no thoughts, but *through a human interpreter* can be made to speak the thoughts of another.

The reaction of non-literates to writing suggests that the user reaction to current intelligent systems cannot, alone, be taken on face value, and that the doubts raised about the Turing Test are reasonable. Humans project intelligence onto the machines, as they once did onto the book. Maybe in a few years others will look with amusement upon those who felt that machines were exhibiting intelligent and autonomous behaviour, as we do upon the non-literates in Gelb's stories. But before we become too dismissive, we must remember that this very 'personalisation' appears to be a function of writing. It occurs through its reifying function which makes of ephemeral spoken words stable, tangible objects, written words. Writing encourages the separation of the text from the author. Indeed, in the interpretation of a spoken utterance the focal question is what the *speaker* meant by the utterance, but with written texts we usually ask what the *text* means. The text gains an autonomy (Coulmas 1989), and artificial intelligence carries this process further.

Even though one may still refer to the meaning and intent of the text, it is important to distinguish this form of expression as literary, and recognize the original source of meaning being with the author, whose intellect is embodied in the text. The program ELIZA, whose creator knew that it had no real intelligence, fooled many of its users who began to engage in deep conversation with it. But, as Bolter has remarked, ELIZA was a "classic example of deferral masquerading as intelligence" (1989: 139). That is, the delayed reaction from the software writer's applied intelligence appeared to the user as intelligent behaviour. Such deferral is characteristic of writing.

Writing and memory

Gaur (1984), in her excellent history of writing, argues that it is primarily information storage. Prior to the development of writing this task was given to a separate group of people entrusted with the task of memorizing those things important to the culture. There are, however, differences between writing and speech. Oral transmission requires personal, and usually extended, contact between persons who are physically present at the same time and at the same place (of course, in the modern era the telephone has broken this traditional concern, but this is not relevant to the current discussion). With writing (in the conventional sense), information is stored mechanically on an object that can be used anywhere, anytime (assuming the object is portable, such as a tablet or book). It can be used by all those able to read and to decode it. Writing gave us impersonal knowledge, as it separates knowledge from the speaker; and also gave a democracy of knowledge, as this text can be read by all with the necessary skill. Memory is still very important, but when one is working with writing a major single effort to learn the rules that govern the script is needed. The information within the book (or parchment or whatever) is then available to all those who have learnt the script, without having to memorize all the book's contents directly.

Early forms of writing are transitional between oral tradition and full writing. They were not very good at representing a full message but acted as a pre-writing, reminding the reader of things (through the reader's own words) rather than specific words (those of the author). For example, the Maori of New Zealand had saw-shaped wooden boards called *he rakau whakapapa*, which were used to keep genealogical records. Youths were taught to recite the name of each ancestor associated with a particular notch in the toothed edge. Similarly much of the knowledge that was required to perform the *Midewiwin*, or medicine dance, of the central Algonquin peoples, and to remind them of other dances and songs of the Mide society, was inscribed on song-boards or birch-bark scrolls (Gaur 1987). As Goody notes, such writing systems can be given a variety of verbal translations, they suggest language rather than transcribe it. There appear to be two forms of this memory aiding function, in one case the writing suggests chants and can only be understood by someone who has already memorized the associated linguistic act. A good example are the birch-bark scrolls, by which the complex rituals of oral tradition were transmitted by the Mide shamans to their disciples. The scrolls were, therefore, secret documents, acting as a mnemonic for the initiate but not communicating to the world in general. In the other case, it is more a prompt, a "peg on which to hang a variety of possible versions" (Goody 1987: 12).

The ability of writing to act as a memory aid has a parallel with the manner in which simple experts systems are actually used in practice. The answers that are given are often less important than the fact that the issues brought up by the system in its behaviour broadly reminds the user of either the things they must do, or the kind of issues they should think about. Often it seems that even if specific answers are given by the systems, they would be of little interest. Consider the following: in the 1960's Harold Garfinkel at UCLA provided a free counselling service to students. However, they had to ask the questions in a form that only required a yes/no answer, and they presented the questions and received their replies without ever seeing the counsellor. Many students felt that they had received some counselling. In fact, the 'counsellor' was merely the random selection of yes/no answers to each query (Collins, Green & Draper 1985). These results throw light on the apparent success of many simple expert systems. They seem to act more as provocations that allow us to be reminded and to help phrase those problems and possible solutions that are on our minds. Similar memory aids are found, for example, in the checklists used by airline pilots. This very term has been used by expert system builders, one of whom:

> in conversation, remarked to me that his system was not really 'expert', more an 'intelligent checklist' (Leith 1990: 50).

The relation between writing an memory has another, and crucial quality. The major advantages of writing are clear. There are limits to what human memory can hold, whereas books can store almost limitless amounts of information. Once freed from the need to memorise information, the mind is able to expend more effort on speculation and critical thought, a point that will be discussed in detail in a later chapter.

Critiques of Writing Technologies

The ability of a 'writing technology' to unburden the human user of the need to remember a great amount of material has been described as both a benefit and a threat. Many of the objections that are currently laid against AI were previously made against writing by Plato in the Phaedrus and the Seventh Letter (Ong 1982: 79). Writing, Plato has Socrates saying, is inhuman. It pretends to establish as external to the human that which can only really exist within. Secondly, writing will destroy memory, making one rely on external means and thus weakening the mind. Thirdly, the text is basically unresponsive: people can give explanations, texts cannot.

Many of the charges against writing were inherited by printing. Hieronimo Sqarciafico argued as early as 1477 that "the abundance of books makes men less studious" (Ong 1986: 28), presumably since one did not have to learn everything but could turn instead to printed reference materials. But the availability of a tool, even printing or writing, does not mean that it will be used — maybe due to the concerns above. The Tuareg of North Africa, for instance, have known writing, but have used it for little else but love letters, charms and occasional poems (Coulmas 1989: 11). Indeed, historically the number of spoken languages that have a written script was quite small, but the success of these languages was due in no small measure to their written character. This suggests that despite certain doubts, the benefits of writing outweigh its failings.

More recently, AI has been criticized on the first two counts: It pretends to make intelligence outside of its natural place in the human mind, and its use threatens the natural intellect of its user — the argument that human users of systems become deskilled. But AI seeks to avoid being guilty of the third, that of being unresponsive, through its expression in the form of a dialogue. As was discussed in the previous chapter, it is traditionally expected that AI systems should be able to engage in a conversation with the human. This thesis, however, is concerned with pursuing the idea of 'AI as writing' and will, therefore, (and despite Plato's fears) adopt this third 'bad' feature of writing and return to its apparently basic unresponsiveness. That is, the 'knowledge medium' proposed later in this thesis will not engage the reader in a dialogue (as understood in the conventional sense).

Writing and Administration

The early writing systems of the Americas had mainly ritual uses, but the situation in Mesopotamia was quite different. The earliest records were not concerned with communication as we would understand it, and certainly not used for setting down myth. The cuneiform writing system had the function of recording the transactions of the bureaucracy — the control and distribution of information, labour, goods, and services:

> Lacking, as yet, any trace of written creative literature from the early archaic periods, we must view the scribe of those times as neither scholar nor poet, but as, above all, a clerk (Green 1981: 348).

Similarly Bottero remarks that Mesopotamian civilisation,

> was quickly caught up in a widespread economy which made necessary the meticulous control of infinite movements, infinitely complicated, of the goods produced and circulated. It was to accomplish this task that writing was developed: indeed for several centuries, this was virtually its only use (in Goody 1986: 49).

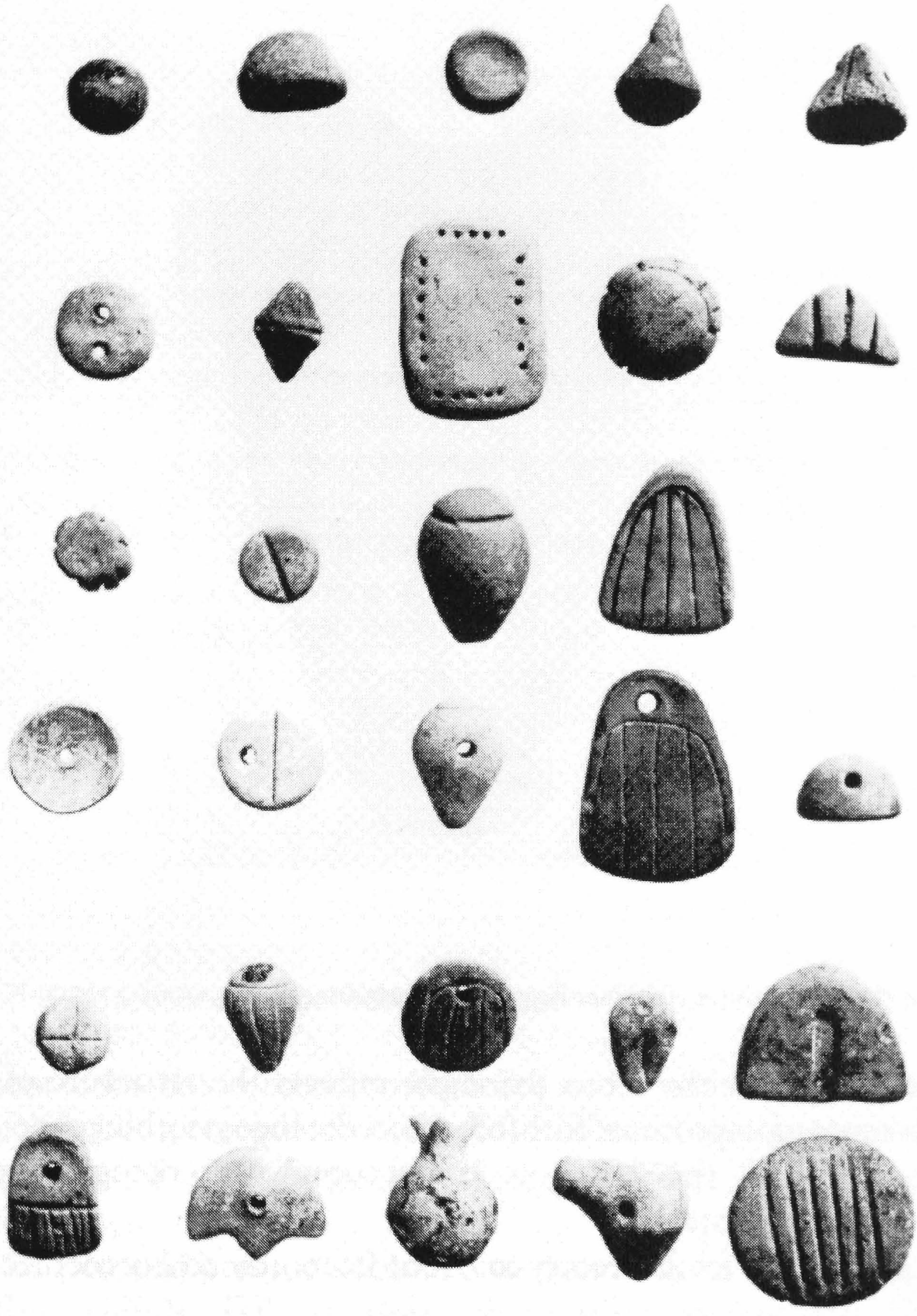


Fig. 3.1 Clay Tokens from Susa c. 3000 B.C.

Recent work by Schmandt-Besserat (1981, 1986) goes further, suggesting that writing has its very earliest roots in accounts. The previously accepted view was that the basis of writing was the picture, and for Egyptian hieroglyphics this appears the case, but Schmandt-Besserat's work concerns the earlier cuneiform. If Schmandt-Besserat is correct in her interpretation (and her results are still controversial, doubts about the dating of the tokens have been expressed — see Haarmann 1990), the earliest precursors of writing would be little stones and pieces of baked clay used for accounting (see Figure 3.1). These tokens were found in a range of forms, representing various economic units: one bushel of grain, one jar of oil etc. To assist in record-keeping, these would be placed in clay envelopes, called *bullae*, which were then sealed. Of course to check the inventory, it would be necessary to break the seal. The breakthrough came with the innovation of pressing the tokens into the wet clay of the bullae to indicate its contents, and thereby avoiding having to break the seal. The significant point of this work was the discovery that the tokens are graphically identical to the earliest pictograms of what became Sumerian cuneiform writing. In other words, the early characters in the Sumerian script came not from painting, but from the tools of the accountant.

Early writing of this administrative form also developed the cognitive tools that were to prove so important for computing: the list, the table and the matrix (Goody 1977, 1986). In other words, there are not just parallels between the two technologies, but in an important sense computing continues and implements in other ways the structures that writing initially created. For example, tree diagrams are common in medieval and Renaissance books. D'Alembert in the Preliminary Discourse to the *Encyclopédie* of 1751 displays all of knowledge as a tree structure (Bolter 1989). What the computer gives are ways to handle such structures in a more complex and useful way, for example, trees can be searched. What the knowledge medium aims to do is to apply, through a computational implementation, other 'structures' that are found within traditional writing, but are currently within the domain of artificial intelligence.

The growth of computing, like that of the written word, can be attributed substantially to its application in commercial and accounting software: accounts payable, accounts receivable, stock control and payroll. AI too, through its origins in computing, shares a general interest in financial affairs.

Indeed, recently, as the more ambitious aims of AI have begun to seem difficult to achieve, it has re-oriented itself towards this domain. The head of one AI company has gone so far as to suggest that "the big question in knowledge-based systems is — Can the customer develop the applications he needs to realise more efficient operation of his business?" (Manuel 1986).

Speech and Writing

One aspect of writing which is of major importance is the relationship between speech and writing. As speech came before writing, there is a belief that it is therefore superior, closer to ourselves. This issue will be addressed later, but it should not blind us the the historical fact that the history of writing has been one that has involved it in a deep connection with speech, be it from its accounting or pictorial beginnings.

Evidence shows that the early attempts to create a language of 'thought' had only limited success, and that real progress came when the link with spoken language was made. Even in picture writing, which one would imagine as having no relation to the spoken language, this process had begun to occur. There is a form of North American picture writing known as 'kekinowin', which is known only to the priests who use it to memorise the correct order and the exact wording of spells (Gaur 1984: 33). In this case, the pictures represent not an idea or object but a sentence or verse. The pictures are in fact read just as if they were a text.

The Yoruba of Nigeria also employ a phonetic aspect in a form of communication. For example, six cowrie shells mean "I am attracted" (to you) because the Yoruba word 'efa' means both 'six' and 'attracted' (Gaur 1984: 33). This is an example of Rebus Transfer which is the representation of a word by pictures to suggest its syllables. The important point that arises from the Yoruba example is the fact that if one does not speak Yoruba, the relationship between the shells and the meaning disappears. It cannot be decoded without linguistic mediation.

The case is naturally clearer in the case of more conventional scripts, Sumerian writing was pictorial, later becoming phonetic as concrete signs came to be used for words that had the same phonetic value, but which could not be pictorially designated due to an abstract or complex meaning. For example, the Sumerian word for both *arrow* and *life* was **ti**. The written word for life could be indicated by using the sign for arrow, with a marker to indicate that is was being used phonetically (Coulmas 1991: 78). This abstraction of the sign from the object to speech was the first step to writing as we know it. Further steps to abstraction were due to the fast drying nature of the clay into which the signs were pressed. It lead to the evolution of increasingly simplified and stylised signs that began to be unrecognisable as pictograms. When the Akkadians conquered the Sumerians they adapted the Sumerian script to their own language. Although the original pictographs were used, the adoption of the phonetic elements lead to further abstractions as the Sumerian sounds bore little relation to the meanings in the Akkadian spoken language. This process was very important, and brought written language away from a thought notation and toward a representation of the spoken language. Even as early as the Old Babylonian period this goal became encapsulated in the Sumerian proverb, "a scribe whose hand matches the mouth, he is indeed a scribe" (Green 1981: 359). The development of writing could be summarised thus:

> ...information which once belonged to the sphere of the scribe's personal familiarity with the language and writing system...became systematically incorporated into the visual writing system itself (Green 1981: 360).

It is, however, another curious phenomena that AI researchers have sought to design a form of description that goes beyond spoken language. AI tries to develop such languages of thought, and faced with a novel script the general population still like to believe that it comes from a purer golden age when man was closer to his thoughts and expressed them directly. For example, before its successful decipherment, the nature of Egyptian writing was greatly misunderstood. It was generally assumed that hieroglyphs had no phonetic value or relation to language. This idea was expressed by Plotinus, a Greek philosopher who was born and raised in Egypt in the 3rd Century:

> When [the wise men of Egypt] wanted to express their meaning philosophically they did not go through the whole business of letters, words, and sentences. They did not employ devices to copy the sounds of a proposition and how it is pronounced. Instead, in their sacred writings, they drew signs, a separate sign for each idea, so as to express its whole meaning at once. Each separate sign is in itself a piece of knowledge, a piece of wisdom, a piece of reality, immediately present (cited in Coulmas, 1989: 61).

Only much later did it become clear that hieroglyphics were used to write the spoken Egyptian language.

Likewise it has often been held the case that Chinese is also 'ideographic'. Leibnitz, for instance, in his quest for an alphabet of human thought, believed an analysis of Chinese characters would lead us in the right direction (Coulmas 1989: 104). But despite popular belief, Chinese characters are not ideograms and the writing thus not ideographic. Characters stand for words, not for meanings (Halliday 1989: 19). The notion of a deep ideographic representation is the dream that AI continues to chase. But it may be a dangerous route to follow. Bloomfield rightly criticized the idea of signs as representing objects as being misleading because,

> the important thing about writing is precisely this, that the characters represent not features of the practical world (ideas), but features of the writers' language (in Coulmas 1989: 62).

Havelock too recognised the development of writing as being towards language as it is spoken :

> For in the beginning they [writing systems] were used to symbolise three different psychological operations: first, counting up and comparing quantities: second, observing physical objects as such in the world: third, the act of naming these objects and the art of relating names and therefore objects to each other. The shapes were used to symbolise these mental acts directly. They went straight to psychological processes in the brain. In a sense they were too ambitious. They were not content to deal solely with phonetics...finally came the systems that sought only this limited aim, of *copying* linguistic noises (1982: 55).

Others argue that Havelock is overstating the case. Writing does leave out aspects of the spoken form, and does not try to copy all the noises. But a connection to language structure seems undeniable.

It is very ironic that AI draws on two cultural drives, that for orality as expressed in dialogue (to pass the Turing Test), and that for a language that is deeper than (and hence not) speech — which seems at first glance to be totally in conflict with the desire for dialogue.

Although the evidence suggests that the development of writing has brought it into close relation with speech, one must avoid making what Harris (1986) calls the "Evolutionary Fallacy", in which the history of writing is seen as one which evolves from pictographic, through logographic, syllabic to phonetic. The implication being in this position that contemporary Western scripts are somehow superior to non-phonetic scripts. Thus, although Egyptian hieroglyphics have a connection to speech, it is not a phonetic alphabet, having many additional non-phonetic features. Many commentators have considered the whole system as cumbrous, and have seen as evidence of a lack of imagination that the Egyptian did not make the final and 'obvious' step of using their existing uniconsonantal signs as an alphabet. Interestingly, during Egypt's Late Period, perhaps under the influence of Greek, the experimented with such an alphabetic system. But these experiments were short-lived, for they seemed to sacrifice legibility for simplicity. The traditional mixed orthography creates distinctive word patterns that enhanced legibility (Davies 1990).

The usual assumption in the West that logographic scripts are more difficult to learn than a phonographic ones. But the facts suggest that the two kinds of script involve different kinds of difficulty. Recent experimental work brings out some of these points more clearly (Sampson 1985). American children who had failed to learn to read even simple English words in two years of schooling managed, in only four hours, to read simple stories written in script that used Chinese characters. In written Japanese, which combines both logographic and phonographic elements, developmental dyslexia is extremely rare. In those cases of acquired dyslexia (where it occurs through brain damage) it affected the reading of the phonographic and not the logographic elements.

Conventionalization

AI tries to get below language so as to represent thought. But the success of writing is attributed to a great extent to the fact that it came to be bound to spoken language. Likewise the knowledge medium, unlike traditional AI, attempts to follow this example by linking the knowledge representation language to existing forms of expression (in this case writing). Just as writing used the existing practice, speech, as a support, so the knowledge medium can use writing for the same purpose. Of course, a support can eventually be abandoned. Harris puts this case most succinctly:

> When that history [of writing] comes to be written ... speech will be seen as the historical crutch on which writing was obliged to lean in the earliest phases, a prop to be thrown aside when no longer needed (1986: Epilogue).

Attempts have been made to throw the crutch away sooner rather than later. One of the most curious, and lesser known, sub-plots in the history of writing is the Universal Language Movement, which arose in the seventeenth century, and whose members

> were concerned with constructing a universal writing system, a real character composed with ideographic signs which they believed directly referred not to words but to things or to notions of things, just as 2 refers to the quality two and not the word two, deux, duo, etc. (Slaughter 1982: 2).

It is interesting to explore the background to the design of these *real characters*, for certain parallels with the conditions that surrounded the birth of AI appear. One major impetus for the development of universal languages was linguistic. Latin had earlier served scholars all over Europe as a virtually international language (Knowlson 1975). Two factors changed the situation. Firstly, there came reports from travellers from beyond Europe. In particular, there was extreme interest in Chinese character-writing, which was, at the time, believed to be ideographic. Secondly, the language situation was beginning to change in Europe, with Latin declining as the use of vernaculars grew. Scholars feared that they would soon be unable to communicate ideas between themselves. It is relevant to note that, more recently, "nearly all computational linguistics, natural language processing in Artificial Intelligence...[has its] origins in MT [machine translation] research" (Hutchins 1986: 19).

The Universal Language scholars aimed to provide a universal character (set of ideographic signs) that through the composition of its signs would accurately mirror the qualities of things in the world and their relations. The language would be knowledge, providing an accurate description of the thing signified. The new language would lead, in Comenius's words, to the "discovery not only of language, but of thought, and, what is more, of the truth of things themselves at the same time" (in Knowlson 1975: 96). It was thought that communication could proceed through direct representations that were representations of the world and not of the speaker's language. A valiant attempt to reverse the dominant trend of writing up to that point. Furthermore, by providing a form of calculus, such as that employed in mathematics, the universal character would need merely to combine symbols to arrive at evident, demonstrable conclusions (Knowlson 1975). AI too, in its construction of intelligent programs, uses knowledge representation schemes which attempt to represent the world, its object and events. "A basic premise of AI is that knowledge of something is the ability to form a mental model that accurately represents the thing as well as the actions that can be performed by it and on it" (Sowa 1983: 2). The difference lies that within the computer both the real characters and the calculus can be achieved mechanically, rather being written on paper with pen.

The importance of these attempts, undertaken by many intelligent men over an extended period, was that they failed. One reason may be found by looking at the case of Marin Mersenne. In his *Harmonie universelle* (1636) he too showed himself to be interested in the idea that one might discover a primitive and natural language that would signify immediately and be without convention. But eventually Mersenne was forced to conclude that, as all known languages signified due to convention, it was not to be expected that any artificial language would succeed where all instituted languages had so far failed (Knowlson 1975). That is, there is no direct and immediate connection between the symbol and the idea, the relationship has to be learnt.

One typical example of such a universal language is that developed by Gabriel de Foigny. He claimed that, in his language, the words could convey knowledge of the objects being signified. The language was constructed through the combination of the five vowels, each combined with a large set of consonants. Each consonant is a sign for a particular quality: C signifies heat, X cold, F dryness, S whiteness, N black and so forth. Foigny was thus searching for a "natural language ... which would signify without convention, and hence be understood by all" (Knowlson 1963: 271). But again, there is nothing innate about the symbol C to signify heat, it is a convention.

It is also curious that the Universal Language Movement saw the example of numbers as being a standard bearer for ideographic representation. For researchers into the history of writing, it is the opposite situation. The use of special signs for numbers is one test of evidence for the link between a writing form and an associated spoken language. For example, three cows are represented by two word signs, one for 'three' and another for 'cow', rather than three identical signs, one for each 'cow' (Goody 1987). It would seem that a language that represented the world out there would have three separate signs for each cow, rather than for the spoken word indicating the word '3', and the word 'cows'. AI representation languages also make extensive use of numbers rather than representing each object of many by a token. It, too, has this link to spoken form.

The knowledge medium approach, used by Running Texts, applies the lesson that Mersenne found so many years ago, resorting to a conventionalising, rather than a mirroring, of reality. A user would come to recognize a set of AI formalisms as representing a form of reasoning, through its association with ordinary text. Ironically, AI might be able to achieve an ideographic quality for its representation techniques, but only through, and after, the power of an existing medium, 'writing', has facilitated conventionalization. Just as writing increasingly throws off its roots in speech, so we no longer need to speak out loud to read and words can be identified immediately as representing an object without concern for their sounds, so AI could come the represent thoughts.

Beyond the need for conventionalization, Slaughter seeks the reasons for the failure of Universal Languages elsewhere. These languages were based on a taxonomic approach:

> The taxonomic episteme on which they were based was one that identified an existing order of things which was, if not always strictly visible, at least apprehensible and always representable. The universal languages were thus predicted on a world picture which existed in a construct of words, in language, or more appropriately for that recently literate age, in a world set down in a book (Slaughter 1982: 217).

That is, we meet a strange turn, where the characteristic of writing to objectify our experience, converting it into discrete 'words' that can be placed in an ordered arrangement, leads to the development of a pure writing totally divorced from speech, in which the world is a series of events that flow into one another. But the design of the various proposed Universal Languages was based on a belief, seemingly mistaken, that the world as revealed through writing, was, in fact, the only nature of the world.

These points indicate a set of design decisions for any putative knowledge medium. Conventional AI insists that the human using the machine should be able to deal with it as in natural conversation. The human should need make no additional effort. Indeed, AI has sold itself as offering exactly this feature, allowing computer software manufactures an interface that anyone can immediately use, giving any particular system an advantage over the competition. But writing systems tend towards uniformity, providing codified conventionalized expressions for the language (Halliday 1989). Agreement has to be reached on the meanings that are to be attached to visual signs. The establishment of such a convention is a form of *social problem* *solving* (Ehlich 1983). Such conventions must be learnt. Hence, a knowledge medium (as a writing) can reasonably presume that the user must learn how to use it, provided that the pay-off for the user is sufficiently large.

Of course, to aid conventionalization, the 'reader' must be able to make the connection between a graphic mark and an idea. Binding writing to speech facilitated this process during the development of writing. By evoking everyday speech, early writing was easier to learn, but could later begin to break away. As Slaughter explains:

> With widespread literacy and standardized spelling, 'the written languages of Europe, which were once purely phonetic, are now to a certain extent ideographic'...Literate speakers, those who read, write and spell, accustom themselves to purely written forms of the language and 'by long habit have formed direct associations between certain familiar groups of letters and the meanings which they represent'. Written words are processed as units and become ideographs for words and morphemes (1982: 80).

Similarly, the knowledge medium can use writing — which its 'readers' already know — to assist in the conventionalization of the AI techniques. For example, the way that a small program can represent metaphorical reasoning can be learnt through its association with the kind of metaphorical text we read daily. Once literacy in AI techniques is widespread, the knowledge medium may begin to be able to represent directly.

Writings leave things out

Although writing has its foundation in speech, there are many aspects of the spoken language that are not represented in the written form: rhythm, intonation, loudness, timbre, pausing, as well as the individual characteristics of one person's speech (Halliday 1989). It is not pure 'speech writ down'. There are two reasons for this state of affairs. Either these features that are not represented in writing are very difficult to write or they are not sufficiently important to require being written.

That a closer phonetic representation is possible, but not employed, can demonstrated by looking at the example of Sampson using a notation devised by Halliday. For certain areas of study, notations are used that represent more information about what is being said that is usual in conventional writing. These notations are very important to linguists and anthropologists concerned with having a record of the subtleties of what is spoken in everyday conversation, for subsequent analysis. In conventional writing these subtleties could be lost. Consider a sentence "Is he sure of it?". The same words can be spoken in a variety of ways that convey different meanings . Halliday's notation (in Sampson 1985: 37) can be used to represent four versions below (// and / represent the boundaries of tone-groups and 'feet' respectively, underlining marks the tonic syllable, and numerals represent the various English 'tones' or patterns of pitch).

//2 is he / sure of it// - rising to a high pitch: a neutral question
//-3 is he / sure of it// - low level pitch rising to mid: 'I'm asking "Is he sure of it?", not that it really matters...'
//5 is he / sure of it// - rising to high, then falling: 'Is he sure of it, because if he *isn't*...'
//4 is he / sure of it// - falling to low, then rising: 'You ask whether he's sure of it? Of course he is!'

Fig. 3.2 Halliday's Representation of Spoken Phrases

These sentences have different meanings, but look the same in conventional English orthography. Indeed, for economy certain forms of shorthand have been developed in which vowels are not written, but readers usually have no problem determining the intended meaning, especially if a context is provided — consider the example *Y mst be crfl wth sch ppl* (Harris 1986: 111).

Another method which tried to symbolize speech as accurately as possible was the Analphabetic Speech of Otto Jesperson (Gelb 1963: 243). In this notation each sound is represented by a series of Greek and Latin letters, Arabic numbers and a few other special symbols. The Greek letter represents the articulatory organ (lip, tongue etc.) and the Latin letter (as exponent) denotes the articulatory position (front, central, back, etc.). For example a sound **u** would be represented as :

$$\alpha\ni^{a}\ \beta g\ \gamma\ni^{j}\ \delta o\ \ \varepsilon 1$$

The system created by Kenneth Pike (Gelb 1963: 243) is even more complicated. A sound **t**, would be represented by:

MaI1DeC*V*veIcAP*paatd*tl*tnrans*fs*SiFS*s

The obvious reason for not pursuing the goal of representing the sounds too closely is the cumbersome nature of any resulting writing. A long narrative written in these forms would be staggeringly large. But there are other, possibly more important reasons. One is that a more exact representation of what is said may not be as valuable as one might imagine. Consider the plural forms of the words: pitch, cat, fog. The final 's', if represented purely by the sounds would have three versions which might be written as 'iz', 's', 'z'. The fact that they are all of the same feature — plurality — is lost. This is why it is better to consider writing as having a relationship to language (langue), rather than the actual sounds spoken (parole). The deeper linguistic fact is easier to represent than the final vocalization which is a product of the ability of the vocal apparatus to perform (consider the difficulty in saying a plural form of 'pitch' using a single 's', as in 'dogs'.)

A similar situation, of the need to write differently from the actual spoken sound, occurs when the spoken sound is the same, but meanings are different. For example, consider the two French language sentences (Coulmas 1989: 45):

> *Elle se lève toujours à cette heure*
> She always get up at this time
>
> *Elle se lève toujours à sept heures*
> She always get up at seven o'clock

In speech they are indistinguishable, but the written forms the two meanings are explicit. A purely phonetic writing would be unclear for the reader. In speech the situation usually aids interpretation, but writing must make more of this explicit.

The important point that can be drawn from 'sounds writings' (such as those of Jesperson and Pike) is, of course, that we do not use them, except for specialized scientific purposes. Writing, as a means of communication, does not need to capture all the elements of what it seeks to represent, only sufficient to allow understanding. It evokes the speech that it represents rather than simulating it. The choice of the visual patterns can be arbitrary:

> Pictorial correspondence between sign form and meaning ('pictographs') might assist the novice, but the fluent reader relies more on mentally-organized information patterns which he has constructed from his experience with the language, the writing systems, and the textual context (Green 1981: 346).

This characteristic has a great bearing on the role of AI techniques as they are to be implemented in a knowledge medium. Such AI programs have normally been considered as being 'cognitive simulations', encoding all the important aspects of a mental performance. If we take the 'AI as writing' metaphor seriously, we can loosen the constraints and demand only that they evoke the line of reasoning they represent, and do not need to provide a scientifically valid simulation of the mental processes that underlie human performance.

But, this issue is complicated by the demands of an effective symbol coding system, one that makes communication more effective. As Salomon (1979) notes, an overriding concern is the degree of correspondence between the external, public, code, and the internal one that ought to be generated in the reader's or viewer's mind. To facilitate communication, the form of writing needs to have less concern with corresponding to the world out there (as the Universal Language Movement intended), but more with corresponding to the internal representation of the reader. One would expect that 'better' communication would occur when the amount of mental translation is less from the symbol system to the internal (cognitive) form, and empirical research supports this view. This point brings the relationship of cognitive science to artificial intelligence back into focus. Although the knowledge medium may not seek cognitive simulations, it should still draw upon the results of cognitive science in the design and implementation of its component implementations. The closer they are to cognitive processes, the greater their power to communicate.

However, although writing does not feel the need to simulate the voice exactly, it often uses its own characteristics to convey meaning. A good example is the way that a question is represented. Instead of representing the tone of the voice rising at the end of a statement, a special mark is used — the question mark. There are other such graphic markers such as the use of italic and capitalization to mark stress and other meanings (Lakoff 1982). Equally, AI (when used as a medium rather than as a tool in cognitive science) does not need to mimic, and certain features of the representation could come to be representations of features of reasoning by convention. There is a trade-off between the need for cognitive complementarity for any AI representation, and the complexity of the representational form.

Other notations employ another means — normal writing itself — to represent those things that the notational format finds difficult, in situations where the notational complexity would be overwhelming. In musical notation words denoting features that cannot be notated are written above the staff: *andante, crescendo, con moto.* One area of written language which undertakes something similar is drama with its directions, such as: [angrily] or [with great condescension]. It is exactly this solution that Running Texts employ in using conventional text for those aspects that would otherwise be hard to 'read'. In addition to acting as an aid to conventionalizing readings of the AI component, the standard forms of written text are also employed to 'fill in the gaps'. The author can use ordinary writing to communicate what is difficult to formalize, possibly awaiting a time when research has caught up and offers a computational technique. One benefit of this incremental approach is that it allows the knowledge medium to be introduced slowly; we do not demand an all or nothing system that can fully represent a significant text.

The Word and the Object

All the points discussed above are specific features of writing that we can exploit to inform the design of a knowledge medium, but underlying them all is a deeper concern, one that is the fulcrum of the debate between orality and writing. How is it that any sign, be it spoken or written, can claim true connection to anything in the world?

Consider again the Golem, often portrayed in AI texts as a precursor — an archetypal automaton. Even within such automata, the 'word' seems to invade the argument. As Knowlson (1975) observes, both Lullist and Cabbalist thought used combinations of letters to reach divine truth, the Golem being composed of the 22 letters of the Hebrew alphabet, which are themselves expressions of divine energy and power. As Gershom Scholem states in his book 'On the Kabbalah and Its Symbolism':

> The secret world of the godhead is a world of language, a world of divine names that unfold in accordance with a law of their own. The elements of the divine language appear as the letters of the Holy Scriptures. Letters and names are not only conventional means of communication. They are far more. One of them represents a concentration of energy and a wealth of meaning which cannot be translated, or not fully at least, into human language (in Knowlson 1975: 83).

What is significant about this passage is that it not only illustrates the potential historical relationship between AI and writing — between automata and the word, but also indicates a major issue that constantly arises throughout this thesis: the idea that these letters and names are somehow deep, not conventional, and somehow truly represent meaning. This notion, that symbols can somehow have a deep connection to the 'thing represented' is one that binds some AI researchers to the Cabbalists.

Curiously, this is but one step in the historical inter-play between writing and speech. Although the Cabbalists may have thought spoken language to be a lesser form of meaning than their symbols, it is more conventional to believe in the priority of speech over any writing.

> The assumed priority of speech over writing goes along with the idea of a pure, self-authenticating knowledge, such that the intrusion of written signs - mere arbitrary marks upon a page - could only represent a deplorable swerve from truth (Norris 1987: 23).

It is this desire for pure representation and for speech-like behaviour that is one of the driving motivations of artificial intelligence. The Turing Test can therefore be seen as a sign of, and a search for, pure knowledge. AI embodies both beliefs, in having a notion of the power of symbols, but at the same time appealing to the more traditional view that speech has priority over writing. It believes in the power of symbols because it (symbolic AI) is based on the notion of symbolic model of mind, in which according to its proponents, strings of symbols capture mental phenomena such as thoughts and beliefs. These proponents argue that the symbol level is a natural functional level of its own and has regularities independent of any specific physical realizations — that is, it is autonomous (Harnad 1990). At the same time AI also believes in the power of speech. The ploy of creating a dialogue implies speech, and speech implies self-authentication.

Searle's Chinese Room

One attack on the Turing Test, an attack that has some bearing on this point, has become almost as famous as the test itself: Searle's Chinese Room argument (Searle 1990). According to the Turing Test, if the computer answered questions, put to it in Chinese, by responding with the correct Chinese answers, then the computer can be said to have understanding of whatever the subject under question was. The Chinese Room argument very effectively questions this. Searle asks that we imagine himself doing all those things that a computer can do. That is, he sits in a room, taking in input Chinese symbols, on bits of paper, and following a set of rules that he has, manipulates them purely on the ground of the shape of the characters, and returns a Chinese language output. The Turing Test says this constitutes understanding, but clearly, Searle, who does not know Chinese, cannot be said to understand the exchange. The manipulation is based on shape rather than meaning. These symbols are interpretable as having meaning, but this interpretation is not intrinsic to the symbol system.

> It is parasitic on the fact that the symbols have meaning for *us*, in exactly the same that the meanings of the symbols in a book are not intrinsic, but derive from the meanings in our heads (Harnad 1990: 339).

In other words, Harnad is making the point that symbol-based AI, the traditional form, suffers the symbol grounding problem. For him, such techniques can act as a medium for the storage and manipulation of symbols that *we* must interpret, as any other medium does, but the techniques themselves cannot understand.

Harnad's solution is to link the symbolic form of manipulation to a connectionist system, which, he argues, acts as the means to ground the representation in reality. Recent years have seen the re-emergence of this alternative AI paradigm, known as connectionism or neural networks. Its proponents differ from the symbolists in arguing that in connectionism cognition is not symbol manipulation but a set of dynamic patterns of activity in a network of nodes with weighted positive or negative connections. Whether this solution is sufficient is not certain. Many believe that it is the essentially biological nature of the brain that creates its form of intelligence. This is not the place to undertake a discussion of these arguments, what is important is the recognition that a purely symbol manipulating computer system relies on an audience to provide an interpretation, it is essentially a medium.

4. Computer-Based Dynamic Documents

The following two chapters deal with a more contemporary technology of the word: computer-based writing systems. These show parallels with work in AI, but more importantly, there is evidence that they are also beginning to converge with AI. Increasingly, to add functionality, writing systems are adopting techniques pioneered by AI, and simultaneously AI systems are beginning to exploit the interface designs used by more conventional writing systems.

The implementation of writing tools on computer platforms has shown the usual development in which the new technology inherits the characteristics of its forerunner. Such a situation is not surprising, as the users of a new technology often need to have clues on how it functions, and call upon their prior experience. It is even common for the term describing the previous technology to inhabit the term for the new, giving a *pristechnolocution*, such as the horseless carriage, wireless telegraph, iron horse and glass teletype. In the same way, the early printers called their new technology *artificialiter scribere* — artificial writing (Hill 1989). But a new technology inherits not just the name but many of the features of its forbearer. Thus an early word-processor looks like a typewriter, having tab markers etc. But increasingly, features of the new technology, the computational nature of the machine, are being employed more directly and visibly for the benefit of both the writer and reader. The developments that are described in this chapter are part of the ongoing exploitation of the new technology. The benefits of the new technology come from the various ways in which it can help writers write different kinds of book, and allow readers to see texts in a new way (Hodgkin 1986).

The question that arises in this context concerns the nature of the new computer-based medium that one can exploit further. Of course, no medium is associated with only one symbol system, and computing can be most effectively used to manage text, images and animations. But it can be argued that while a medium has numerous symbol systems at its disposal, some of them are more essential to the medium than others (Salomon 1979). For example, television could be used to broadcast only printed material (scrolling past like the credits at the end of a film), but it could not be considered a full use of the forms available to it. Similarly, the use of computing to reproduce and imitate existing forms can only be a stepping stone to a situation where the dynamic, symbolic, nature of the computational process itself is employed in the medium. This is occurring, and increasingly computation has been applied not only to increase the ease and speed with which a document can be created and edited, but to transform typography itself, particularly those typographies that represent the text's rhetorical structure, and lend themselves more easily to formalisation.

Computer-Based Extension of Paper Methods

Although many computer-based writing systems are based on the design and ideas already extant in paper methods, the computer based version invariably begins to offer new functionality. The computer as a word processor is very different from writing by hand or even from the mechanical device of the typewriter. One can edit by selecting whole sections of text, words, sentences and paragraphs, which can then be deleted, copied or moved elsewhere. Certain word processors, such as Word, also offer the writer an *Outline Processor*. This allows the writer to make the traditional written outline dynamic. Topics can be promoted or demoted, shifted from being under one heading to another. The whole text can be restructured at a totally abstract level, independent of the base text. However, the source for this new functionality can be easily identified within the writing task as previously undertaken. So, although a typewriter cannot edit sentences or paragraphs, the cut and paste facility offered by almost all word-processor is clearly an automation of the manual task previously undertaken with scissors and glue. Similarly, the outliner is a more efficient version of what is possible by manipulating separate pages. The final document could have been created with the machine or without it, a reader with a printout would have difficulty in knowing which system had been employed. The real innovation comes when the active nature of the machine becomes an intrinsic part of the text as written. Exploiting this dynamic characteristic of the machine creates a new medium, to do otherwise is to waste its resources. It would be as if the inventor of the cine camera proposed it as a means of getting lots of different stills of a complex and constantly changing scene, from which the photographer could select the best to print up.

Enlarged & Knowledge-Based Spreadsheets

Spreadsheets offer an interesting example of the way that the structure inherent in writing can be made dynamic. They take the tabular form that has its origins in the nature of writing, and activate it. The *table*, like all texts, is a symbolic structure and can be subjected to symbolic manipulation (Bolter 1991). That is, one does not see a spreadsheet as a set of static numbers. These values are active, and can be calculated and recalculated, giving spreadsheets their power. Spreadsheets are, however, not just calculators. The numbers and the equations do not stand alone. To give them meaning to the reader or author, conventional textual forms are placed in certain cells. Thus the typical business spreadsheet has the names of the months of the year, or terms such as *sales* or *profit*, and the spatial layout of the spreadsheet creates a link between executable and non-executable components. The text adds to the interpretation in way that would be difficult with the equations alone, comparable to the way that text is added to music notation as described in the previous chapter.

One can identify a number of developments from the simple word-processor or spreadsheet. Firstly, extensive additions to the non-executable (the textual) component, to place the calculations within a larger context. This is achieved through the development of the integrated packages that combine spreadsheets and word-processors. The text can now surround and envelop the calculation, for example, the ClarisWorks package. The second, and possibly more interesting, direction has been the attempt to make the symbolic, textual aspect executable by employing symbolic computation. The work of Kosy & Wise, or the FINANZ system, for example, in which the text can be generated by AI techniques to supplement the numerical component. It is exactly this process that the linking of text and AI mirrors, but spreadsheets, being based on older and better understood techniques have achieved this integration earlier.

Clarisworks

Clarisworks is one of the new range of integrated packages in which text, spreadsheet and graphics elements can be woven together in one document. If they desire, the authors can merely use the package as a means of creating a document that is printed on paper and distributed to the readers. But this wastes the potential available in the package. If, on the other hand, the document is delivered on computer readable media, such as a floppy disc, the reader can read it through the package. This gives the reader the potential of exploring the document in a way difficult in a paper version. For example, take a screen image from Clarisworks:

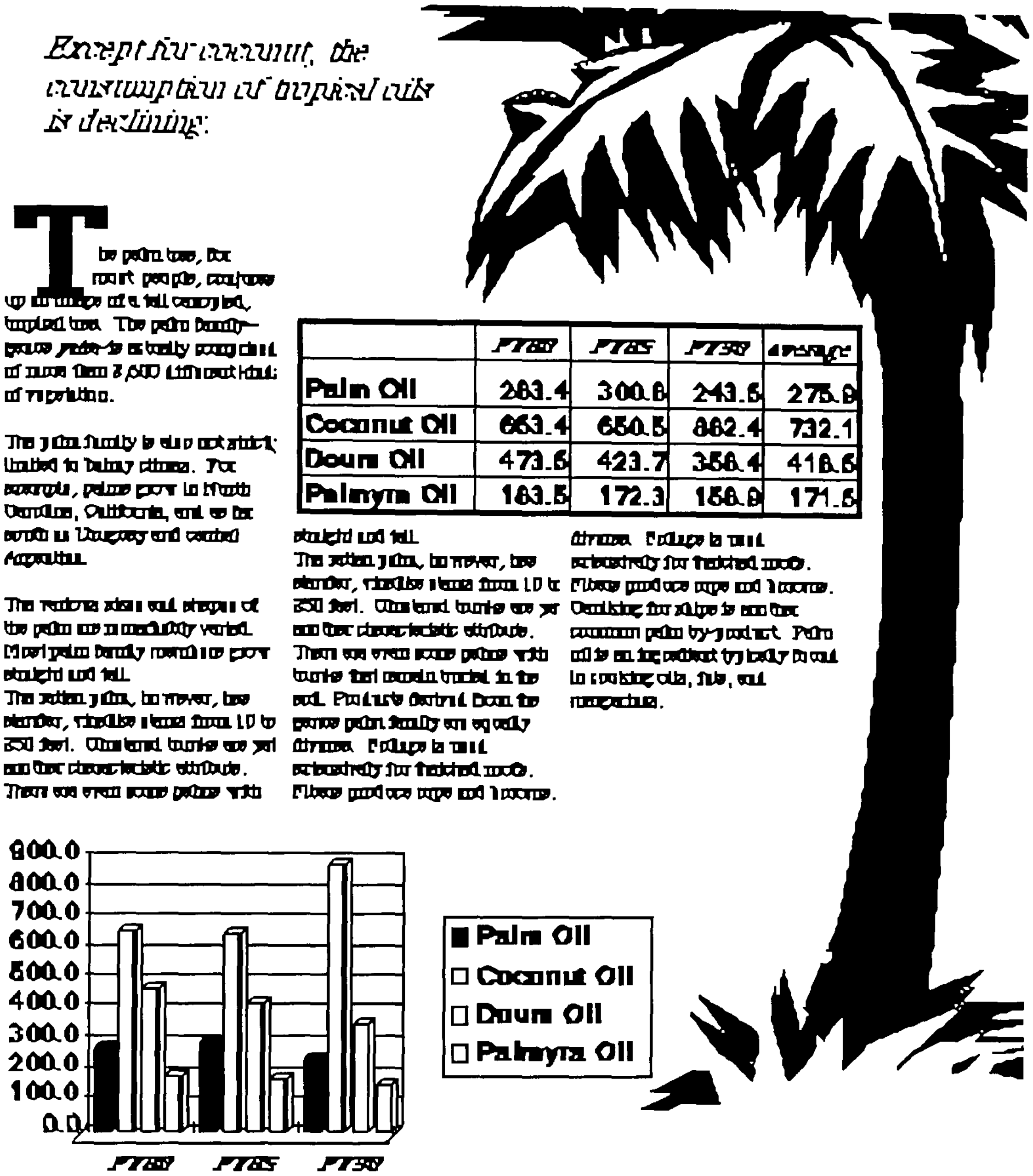


Fig. 4.1 Screen Image from Clarisworks

'Readers' using the on-screen version can, for instance, modify the values in the spreadsheet to suit their own beliefs, possibly changing an expected inflation rate on a long term financial plan. A reader could also appropriate the document and update the sales figures, maybe based on revised information. The spreadsheet will recalculate automatically, and redraw the graph on the bottom left. The reader can then, if need be, edit the text to match the changes. Such appropriation of a text is difficult if not impossible in a traditional paper-bound document.

In the Clarisworks example, the reader (who can now also be an author) would have to edit the text separately. Other systems offer the promise of automatic text generation. The work of Kosy and Wise (1984) is an excellent example of the close integration of spreadsheet and computationally derived text. It was designed with a more conventional view of AI systems in mind, that is, it answers questions that have been posed by the user. But as we shall see, it is not difficult to modify the idea to make it become a dynamic document creation system. The way the system works is best explained through an example. Kosy & Wise have a financial model for an imaginary company, called The Battery Company, which is expressed in a spreadsheet, whose tabular display looks thus (1984: 177):

	1972	1973	1974	1975	1975
Volume	100.00	120.00	132.00	145.20	145.20
Selling price	35.00	35.00	36.40	37.86	39.37
Gross sales	3500.00	4200.00	4804.80	5496.69	5716.56
Labour/unit	9.00	9.00	9.36	9.73	10.12
Matl.Price/unit	8.00	8.00	8.64	9.33	10.08
Material/unit	8.00	8.00	7.34	7.93	8.57
Shipping/unit	2.00	2.00	2.08	2.16	2.25
Unit cost	19.00	19.00	18.78	19.83	20.94
Variable cost	1900.00	2280.00	2479.49	2879.19	3040.42
Indirect cost	285.00	342.00	371.92	431.88	456.06
Production cost	2185.00	2622.00	2851.41	3311.07	3496.49
Gross margin	1315.00	1578.00	1953.39	2185.62	2220.07
Operating exp.	415.00	630.00	720.72	824.50	857.48
Interest exp.	0.00	0.00	0.00	0.00	0.00
Depreciation	35.00	35.00	35.00	29.00	29.00
Mgmt. salary	182.00	236.60	246.06	255.91	266.14
Overhead cost	632.00	901.60	1001.78	1109.41	1152.63
Profit	683.00	676.40	951.60	1076.21	1067.45
Profit margin	16.00	16.00	17.62	18.03	18.43

Table 4.1 Spreadsheet for the 'Battery Company'

The financial model which underlies the spreadsheet is a series of equations that relate the components of the model. The values in many of the cells are calculated by applying the equations to values in other cells. Examples of the sort of equations within the Battery company model are:

> gross sales = volume * selling price.
> gross margin = gross sales - production cost.
> production cost = variable cost + indirect cost.
> operating expense = 0.15 * gross sales.
> inflation = 0.04.

Kosy and Wise have implemented an algorithm that uses the equations, and a simple text generation mechanism (which is template driven), to generate text explanations from the model. The user can now ask a question such as: *Why did profit go down in '73 even though gross sales went up?* The system then generates from the financial model a readable reply (Kosy & Wise 1984: 180):

> Profit went down in 73 primarily because overhead cost went up and profit = gross margin - overhead cost.
> Although gross margin went up by 19%, overhead cost went up by 41% and the latter outweighed the former.
> Gross margin went up in 73 primarily because gross sales went up and gross margin = gross sales - production cost. However, the increase in gross sales was not enough to affect the change in profit.
>
> Overhead cost went up in 73 because operating exp and mgmt salaries went up and overhead cost = operating exp + interest exp + depreciation + mgmt salaries. The increase in overhead cost was due primarily to the increase in operating exp. However, the effect of mgmt salaries was also significant to the change in profit.

What is interesting about the work of Kosy and Wise is that although their examples are numerical, they employed typical AI approaches in the generation of the explanations. They asked a human analyst to explain spreadsheet results, and from the transcriptions of these protocols they derived a number of cognitive features that are implemented in the explanation algorithm:

- distinguish the relevant parts of the model from the irrelevant.
- distinguish the significant effects from the insignificant.
- translate quantitative information into a qualitative characterisation.
- summarize if the same reason accounts for more than one result.

It is also noteworthy to compare the Kosy and Wise explanation from the table, with a simple verbal description of its contents. In the following chapter we will briefly look at the power of lists and tables, as enabled by writing technology, to make certain regular information (such as train timetables) clearer than a verbal description of the same information. In this case, we find a curious turn: Relationships between the data, there in the table and equations, are now made clearer through a more textual, expository form. A very important aspect of the relationship between free text and structured tables is that that the verbal form only makes sense when it answers a question. Representing the material in the spreadsheet fully through using the Kosy and Wise algorithm would produce volumes of text. (The significance of specific questions to be the source of text-like explanations is another feature that Running Texts take up: question and answer relations become crucial.)

A similar integration of spreadsheets and knowledge-based systems has been undertaken by Fischer & Rathke (1988). Their FINANZ system also allows explanations in textual terms of the contents of specific spreadsheet cells, and although the system is not as extensive as that of Kosy and Wise, it offers other features such the resolution of conflict caused by several constraints.

Although the Kosy and Wise design was for a dialogue system, to be used in a conversational mode, one can easily visualise how the system could be modified to be a writing system, an extension of a package such as Clarisworks. An author using the system could create document embodying a financial analysis and plan. The explanation would be part of the document along with the spreadsheet itself, and conventional text could also be added to provide additional material that cannot be derived from the computational component. When a reader now comes to modify the spreadsheet model, not only will the graphic change to take the new values into account, but the textual explanation paragraph generated by the Kosy and Wise technique will also change accordingly. Parts of the text itself has become activated.

Mathematica

Another system which extends the ideas found in Clarisworks in another direction is Mathematica, which allows the author to create documents that embed within them mathematical equations and associated graphs within an explanatory text. The text is not computationally generated as in the Kosy and

Wise example, but the range of mathematical functionality far exceeds that of a simple spreadsheet.

Mathematica can, if required, be used in a simple interactive (dialogue) manner, where the user types in mathematical queries, which the system can solve. But like Clarisworks, it comes into its own when it is used to create 'live textbooks' called *notebooks*. These combine the mathematical capabilities of Mathematica with explanatory text. The reader can read the text in a notebook to learn about a subject, and then use the Mathematica definitions within the notebook to do calculations and, where relevant, to produce graphical images from the results (Wolfram 1988).

A good example of the use of Mathematica is the paper by Slaney (n.d.), a technical report which describes and implements the computational model of the cochlea developed by Richard Lyon. It is available as both a printed research paper and a Mathematica notebook on floppy disk. The notebook is a combination of words and equations that can be printed or read on-screen like the normal paper. However, in addition, it allows the user to interact with the equations (in bold), changing parameters, and to replot the figures (such as that at the bottom of the Mathematica screen image below).

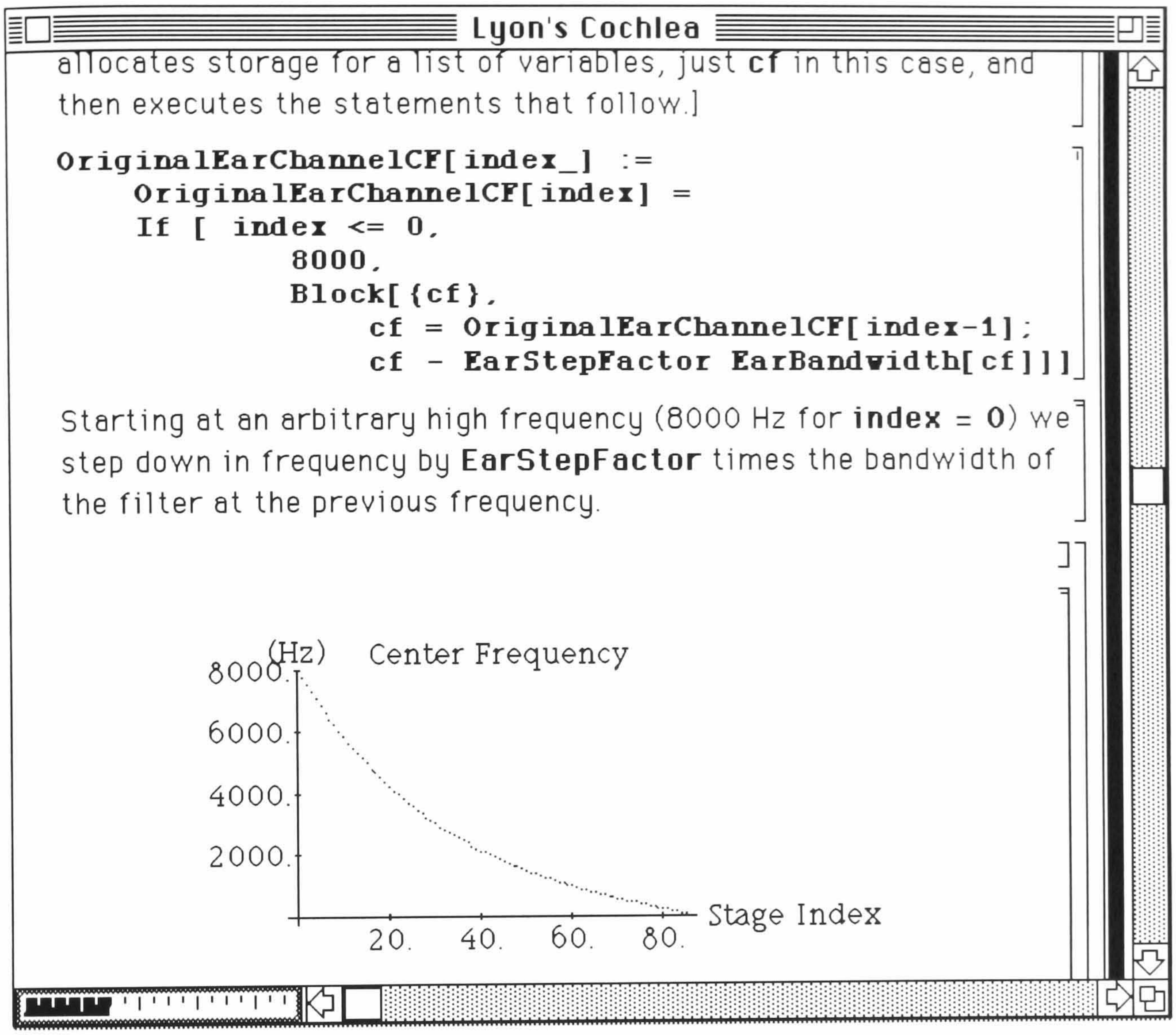


Fig. 4.2 Screen Image from Mathematica

Computable Hypertexts

In addition to being used to add to word processor packages, notions of computability are also being imported into hypertexts, a new generation of computer based text systems. These will be discussed in more detail in the following chapter, but briefly, hypertexts can be described as follows:

> The concept of hypertext is quite simple: Windows on the screen are associated with objects in a database, and links are provided between these objects, both (graphically) and in the database (as pointers) (Conklin 1987: 17).

In effect, hypertext systems appear to the user as a window, either of text or image. Certain elements (textual or graphical) of this window are 'hot'; when selected by the user another window appears, a window to which the hot area was linked in the database. The HyperCard package for the Apple Macintosh is an archetypal hypertext for personal computers, and KMS (Akscyn, McCracken & Yoder 1988) is an example of a larger, workstation-based, hypertext. The major difference between hypertexts and conventional texts is their supposed non-linearity (an important issue to which we will return later).

Naturally, as hypertexts require a computer as a platform, the idea of importing computational components into hypertexts, as has occurred with the linear texts above, was a simple and obvious step, and has been implemented by Schnase and Leggett (1989). An individual energetics model for Cassin's Sparrow was created, in which both the computations and the intellectual work of the research were performed in a unified hypertext environment. Field data, computations for data reduction, tables, figures and text for an article submitted to the journal *Ecological Modelling* were integrated in a single hypertext. Used in its hypertext mode, the 'document' allows the reader to modify the weight of a bird, and the number of nestlings. The simulation can then be run again. The final hypertext can also be e-mailed to any location for other scientists to comment upon and modify the work, creating a community of collaborative workers. Schnase and Leggett (1989: 192) make the point that "[t]he usefulness of hypertext in the sciences *depends* on its computational capacity".

More recently this idea has been extended from straight mathematical computation to executable Prolog representations of engineering norms (Schwabe, Feijo & Krause 1990). The structure of the norms is embedded in both the hypertext structure and Prolog programs. The user is thus able to execute the norms, during which the system will engage in a dialogue to obtain the specifications of the entity to be evaluated. A typical example is to verify if a pipe satisfies certain restrictions imposed by the norm on the stresses that it must withstand. The structure of design code can be represented by an And/Or graph in which provisions (norms) which must be met by calling sub-provisions. Thus in the example they give (slightly adapated to remove a small inconsistency between the diagram and the program), Norm NB3000 can be met by meeting NB3100, and either norms NB3600 or NB3200. Such sets of norms can be represented either graphically :

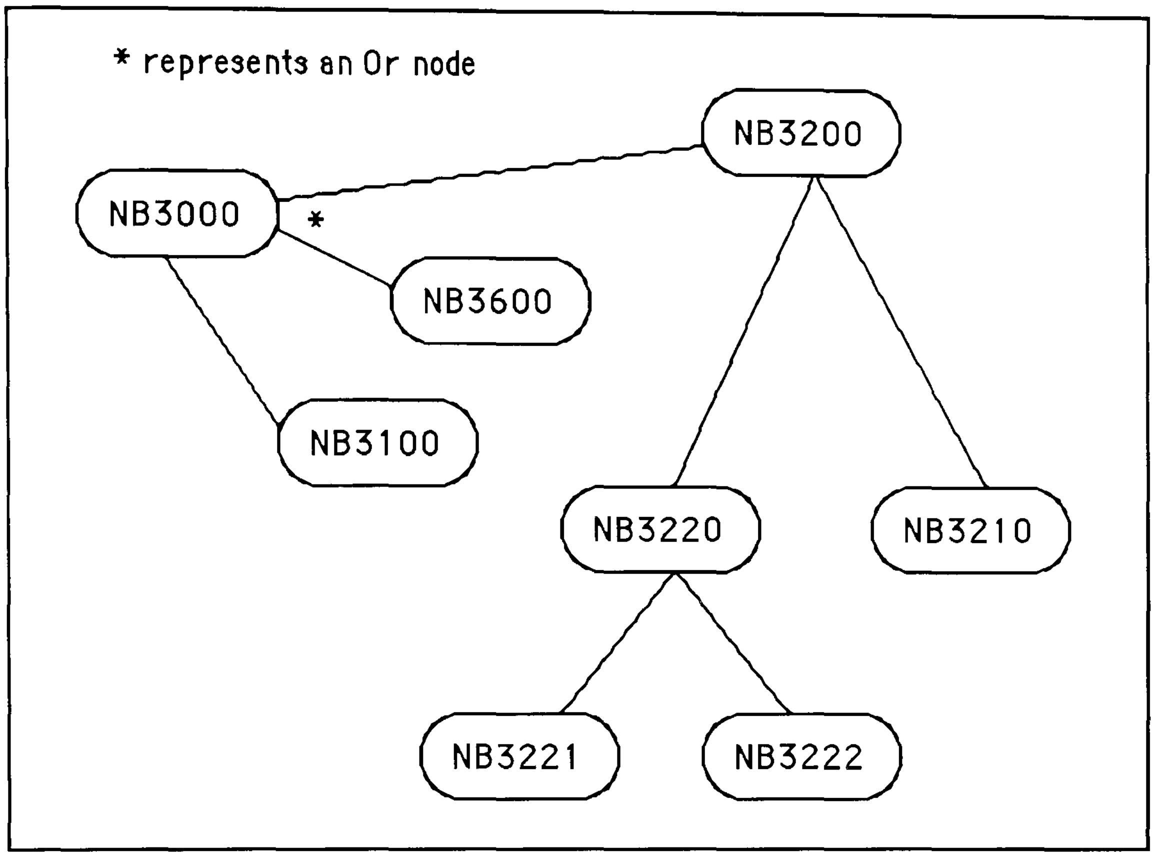


Fig. 4.3 And/Or Graph for Engineering Norms

or as a logic program:

```
nb3000(Entity) :-
   nb3100(Entity),
   nb3(Entity).
nb3(Entity) :- nb3200(Entity).
nb3(Entity) :- nb3600(Entity).
nb3200(Entity) :-
   nb3210(Entity),
   nb3220(Entity).
nb3220(Entity) :-
   nb3221(Entity),
   nb3222(Entity).
```

Fig. 4.4 Logic Program for Engineering Norms

There is an explanation facility available, created by successful running of the logic programs, which shows the user the path of nodes that lead to the successful execution. That is, the system generates a text component. The Schwabe program thus has the same feature as the Kosy and Wise spreadsheet explanation system: there is an intimate connection between the computational component and some part of the text.

The systems that have been described above indicate a certain direction in which computer-based text systems are heading, and suggest various design aspects for a knowledge based writing system. Mathematica creates documents by making a mathematical vocabulary available to the author and reader, and these are assembled, together with explanatory text, for those aspects that cannot be mathematically formalized, into a dynamic document. The work of Kosy and Wise indicates how algorithms derived from areas traditionally ascribed to artificial intelligence, can contribute to this vocabulary. The next step in our argument is to address the question of the nature of the AI techniques that can be assembled together to form a document. Previously we have used the metaphor of these being an alphabet or vocabulary from which the author creates a text. Mathematica and its provision of a notation for certain mathematical functions provides another. The following section will explore the work currently being undertaken within AI that appears to point to such a vocabulary or set of functions.

Distinct Methods in AI

The earliest examples of programs in the modern tradition of AI can be dated back to the late 1950s. This early work was based upon the notion that there were a small number of significant underlying principles that characterized intelligence, much as Newton's Laws were the basis for a physical description of the world. This work had some early successes in its application of this approach but problems began to appear: in particular, programs which worked on one area would fail on another. A better solution came with the second generation of AI systems. It was found that the general strategies, although necessary, were not sufficient. What was also required was a large amount of knowledge, for

> no matter how natively bright you are, you cannot be a credible medical diagnostician without a great deal of specific knowledge about diseases, their manifestations, and the human body (Feigenbaum & McCorduck 1983: 38).

In other words, to make the programs perform adequately it was necessary to 'write' into the program facts and domain-specific heuristics. Such programs, where real-world knowledge was represented, are known as knowledge-based systems. It was this step in the development of the technology that leads us into the realm of writing. Some very early programs had tried to mimic the behaviour of brain cells, but these knowledge-based systems required the programmer to type in symbols, 'words' that represent the facts and structure of the world - that is, to 'write the knowledge down'.

> To summarize: an early belief in knowledge-based systems work was that large amounts of knowledge could be coupled with simple inference schemas to model expert problem solving (Kornell 1987: 208).

It is programs of these sort, known as GOFAI, for Good Old-Fashioned Artificial Intelligence, to which the ideas of this thesis apply.

Generic AI Methods

More recently, AI has been moving away from using a small set of simple inferences over a large knowledge base, and towards the creation and use of a number of distinct AI methods. That is, the AI techniques to be implemented are not a few general methods intended to solve all problems, nor just a bundle of facts and rules, but a collection of so-called interpretation methods or generic methods. There seem to be various *patterns* of reasoning, which have until recently been neglected by the knowledge engineers. For example, many expert systems, employ a method known as heuristic classification, which "characterizes a *form of knowledge and reasoning*" (Clancey 1985a: 290). Kornell (1987) also recognizes these 'patterns of reasoning' and criticizes many knowledge engineering tools for not providing a wide enough range of reasoning types for the systems developer to call upon. All too often the choice of reasoning employed in any final system results solely from the nature of the tool already available in the organization. Bylander & Chandrasekaran (1986), in particular, have urged that systems builders should identify generic tasks — basic combinations of knowledge structures and inference strategies — to be the components for any final systems. Their reason for adopting this approach was a recognition of the 'interaction problem'. The conventional wisdom in expert systems development was to keep the knowledge base and inference engine separate. But the interaction problem identifies that knowledge representations have a close relationship with how they are used to solve problems.

The solution proposed by Bylander and Chandrasekaran is to exploit this effect and identify generic tasks, which are characterised by information on:
- the type of *task*, what is the generic task good for?
- the *representation* of knowledge. How is the knowledge organized to

accomplish the task?
- the *inference strategy* that is applied to the representation to achieve the task.
- the types of *concepts* involved.

Some example of generic tasks that they propose are:
- *Hierarchical Classification*: given a description of a situation, determine what categories or hypotheses apply to the situation.
- *Object Synthesis by Plan Selection and Refinement*: design an object (which can also be an abstract object e.g. a plan) to satisfy a specification

A similar approach is being taken by the research group developing the KADS knowledge acquisition methodology. Their motivation is to assist knowledge elicitation, providing a set of models for typical expert system performance tasks that can guide the expert systems designer with a specific project. They provide a library of interpretation models which are descriptions of the knowledge that is required to perform a particular class of tasks (Breuker *et al.* n.d.). The VICE system (Huuskonen 1992) is also based on a set of explicit problem-solving tasks.

More recently, the notion of seperate and generic methods has become one of considerable research interest. Due to the effort required in the construction of knowledge-based systems, it has become clear that having re-usable components could assist in a cost-effective fashion. One major effort is the 'Shared, Reusable Knowledge Bases' Working Group of the 'Knowledge Sharing Effort' (Neches et al. 1991; Patil et al. 1992). Some within this working group are looking at the specification and development of off-the-shelf knowledge components that can be re-used. Amongst others, they are currently exploring standard components for qualitatitive reasoning, the design of modular reasoning mechanisms, and the specification of the semi-formal hypertext representations (such as gIBIS, described in the following chapter). Similar work is also being undertaken in Europe: Steels (1992), for example, is designing systems using an componential approach where reuasbale chunks can be assembled to create new applications.

The importance of this componential approach becomes clearer when one considers most media. They consist of a assembly of components. Thus writing is a series of letters and words, cinema is a series of short image/sound sequences and so on. What allows us to build a knowledge medium is the ability to do the same with the many AI techniques. The generic task or interpretation method approach offers the kind of units that we seek. There is one further refinement that we require, however, for actual use. A generic method needs to be instantiated, that is a specific query over specific data, employing the inference strategy. To communicate anything about anything these must be complete. Therefore, the basis of the Running Text system is the inference method, but associated with it is the query and model data required.

Before leaving the computational documents, a further, crucial, lesson can be drawn from Mathematica. As with writing generally, readers and authors are willing to learn the notation if the benefits outweigh the effort required to learn it. No professional engineer would expect to proceed without a firm knowledge of the mathematical structures needed. It can be argued, therefore, that user of a knowledge medium would be willing to learn the mechanisms that underlie the various generic methods of AI, in order to be able to use them effectively.

5. Writing Technologies as Structure

The imposition of structure on words is a process that goes back in the history of writing. This chapter will discuss the development of the structuration of text, in particular it will describe contemporary hypertext systems which use the computational resources of the machine to structure text dynamically. These systems use techniques that have a close relation to AI methods, and provide the final step in argument which leads to the design of Running Texts.

List, Tables and Trees

The history of writing shows an increase through time in the use of text structuring mechanisms to contribute to readability. Goody (1987) has investigated this aspect in detail and shows how the list, table and tree are inherent in writing, due to its graphical, spatial, nature. These written forms take category items out of the sentence structure and group them by similarities. Moreover, the individual boxes tend to be exclusive, fruit here and vegetables there. As with other forms of writing, the very idea of 'contradiction' becomes sharpened.

If we compare the purely verbal, written down, form with the structured spatial form encouraged by writing, it is not difficult to see the benefits. For example, take the following descriptions of family relationships:

> My mother's (Sylvia) parents - called David and Ellen - also had three sons: David, James and Kenneth, who married Rene, Sheila and Margaret respectively.

The content can be equally represented in a tree form, which is easier to understand as the structural relationships are made more explicit:

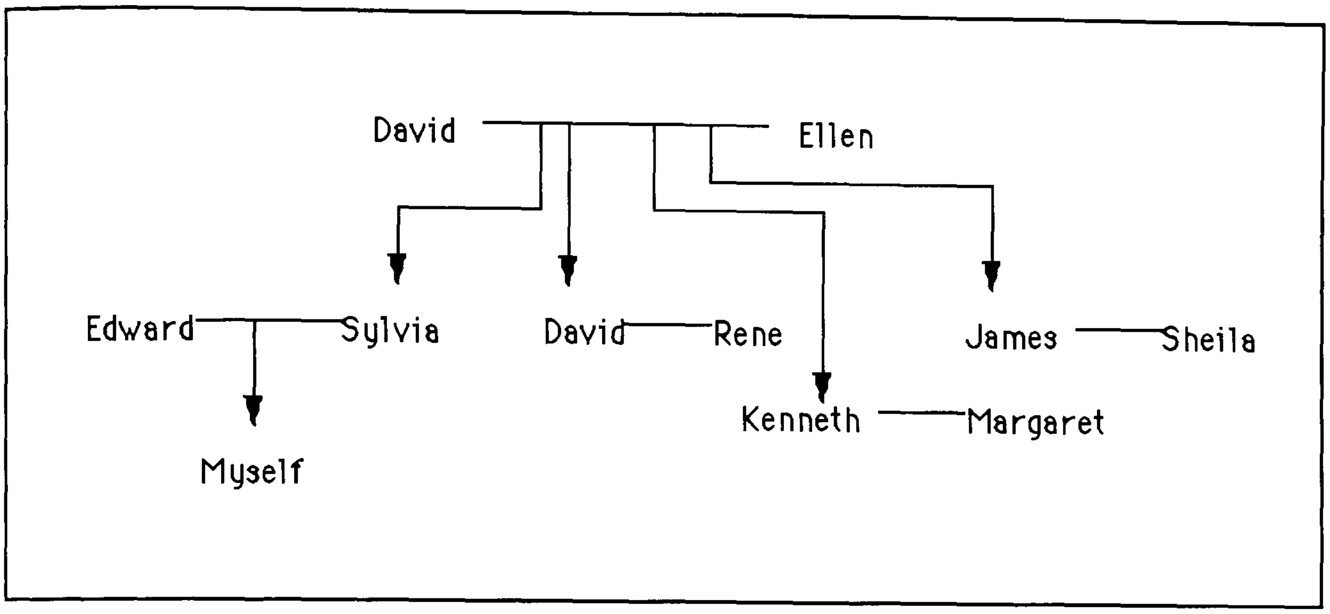


Fig. 5.1 Diagram of Family Tree

In the case of a matrix the clarity gained is often more obvious. Imagine a verbal, spoken, form of a train timetable that would have to be remembered in an oral culture. To find out the information that a traveller wanted would entail remembering the whole oral text, reciting it until the the information came up.

Computing now makes the structures inherent in such graphic forms — the list, the table and the tree — available. One example is the writing support tools developed by Neuwirth & Kaufer (1989). They provide a Synthesis Grid to which the author can automatically route notes. It contains a number of cells, and has the appearance of a textual spreadsheet. Like grids in writing, we see that certain cells are empty and the reasons for this need to be investigated. They also provide a Synthesis Tree, which allows the author to structure ideas in a tree form to see relations more easily. From this starting point, the tables, trees and lists can be used as data structures, formal structures upon which the computer can operate.

Punctuation

Another form of structure that writing adds to text, a form not found in spoken utterances, is punctuation. Punctuation conveys content as much as the words themselves. The division of written text into words and sentences, which we take so much for granted, developed only slowly (Gaur, 1984: 55; Bolter, 1989). The majority of ancient scripts — Egyptian, the cuneiform of Mesopotamia and many of the Indian scripts — did not separate words or sentences. In these societies the scribes were so immersed in the writing conventions that they did not need these aids to legibility. Similarly, written Japanese did not employ punctuation until after exposure to Western writing in the late nineteenth century (Twine 1984).

Greek alphabetic writing also began as a string of letters, without spacing or punctuation. Over time a number of innovations were introduced:
1. The line direction was standardised.
2. Spaces were placed between words.
3. The stop was introduced between sentences.
4. Upper and lower case letters were used systematically, the capitals given special functions.
5. Special symbols were introduced to indicate links, interpolations, and omissions (hyphen, parenthesis, apostrophe).
6. Other, more detailed punctuation marks were added: comma, colon, semi-colon, dash.
7. Further marks introduced: quotation marks, question marks, exclamation marks (Halliday 1989: 32-34).

One aim of these was to represent the natural units of the language in writing. Another function was to specify its speech function — is the sentence a statement or question? The question and exclamation marks specify very clearly what a full representation of speech would find very difficult. One can conceive of using Halliday's notation to represent the rise in tone at the end of a spoken question, but the written form achieves this far more effectively with the simple symbol '?', although it does have to be learnt (not a demanding task, however). It appears that writing seeks to overcome the criticism that it is not speech by trying to convey those features found in speech. But, as already described, writing does not merely try to simulate the spoken form, but employs its own characteristics, its spatial and graphic nature to convey information.

The addition of structure, through graphic forms and punctuation, greatly assisted the text's readability. Thus the first alphabetic writing, which consisted purely of a string of characters, had to be read out loud for the meaning to be extracted. The development of spaces, upper/lower case, and punctuation marks allowed the author of a text to represent more of his intentions (Halliday 1989: 33). Although medieval copyists were not supposed to make any changes to the text, they often did, and not only through error. They also began to add punctuation. For "[b]efore Charlemagne, there were no spaces between words, no periods, commas, semicolons, or explicit paragraphing" (Troll 1990: 111). Troll also cites Murphy who argued that the scribes "punctuated...where they thought that confusion was likely to arise in the minds of the readers for whom the text was prepared" (1990: 111). Increasingly, the intention of text, gained by the reader through a shared background with the author, is made more explicit, as texts targeted a wider audience.

Typography

Later, print continued this process. Typography allows the author to convey further a sense of what is intended, through the use of font size and type, paragraphing, titles, footnotes (which are often 'asides') and so forth. Waller (1988) describes typographic ideas that were never adopted, but continue this line of development:

• square span typography: in which phrases were grouped in small stacks, thus:

Andrews proposed	what he termed 'square span' typography	in which phrases were grouped in small stacks

• *spaced typography*: in which spacing of phrases was important.

In spaced typography,	extra space is added	between 'thought units'

• and *semantic line breaks*: used by Gerstner in his integral typography, where line breaks were placed to break the text into semantically distinct components, for example:

How is 'integral typography' expressed in this book? First by dividing the material into examples and commentary. What can be said more effectively in an 'illustration' than in words is said in an 'illustration' and vice versa. Examples and commentary alternate throughout, that is, are mutually illuminating - as exploration requires if it is to progress.

In a similar vein, Jewett (1981) advocates the use of multi-level indentation. The various parts of the text are classified as being either: main ideas, elaborations, or parenthetic remarks. The main points are placed left justified on the page, with three 'highlighting' lines next to them; the elaborations are indented one tab, with two lines beside them, and the least important, parenthetical remarks are indented two tabs and have one line beside them. He claims that this organization facilitates reading.

Interestingly, spaced typography has an historical precursor in an Old English manuscript: the Tollemache manuscript of King Alfred's translation of Paulus Orosius's *History of the World*. In this manuscript scribal spacings show a great variation (though not of the size used in spaced typography). Examination of these (Cyrus 1971) has shown that they represent certain linguistic features that were eliminated or normalized after the introduction of printing. Examples that readily stand out in the manuscript are: articles, prepositions, the negative particle, pronouns, determiners, and conjunctives are frequently not separated from the following 'word'. On the other hand, one eighth of all spaces occur within 'words', although these are usually between morphemes. Spaces also often occur in compound words separating the prebase, postbase and prefix from the base. Inflectional suffixes are, however, almost never separated. At this level the spacing represents prosodic features. On a larger scale, the variation in spacings mark major syntactic units or clauses, presumably due to the relation between prosody and syntax.

Recently, modern computer technology has allowed for the simple re-introduction, and evaluation, of spaced typography. As part of an empirical study to evaluate its readability, Bever et al. (1990) used computer programs that assign extra spaces between word groups according to major phrases. Results show that the readability of texts could be improved by using such an automatic parser. The results also suggest that a single clear separation gap is more effective than a variation of separation for different types of relation as used in the Old English manuscripts. Furthermore, they also indicate that the Old English prosodic layout, in which the spoken form is used as a guide, is harder to read than normal text, possibly because it calls upon auditory processing strategies — a result which has some bearing on the discussion of the connection between writing and speech. This result could be explained if writing has begun to leave speech behind, and is processed seperately in highly literate individuals.

If we were to use a parser to space text on screen (or prior to printing), computer technology would (partly) return us to a pre-printing form of 'typography'. Extending this point further, it is a curious thought that had technology developed otherwise, and printing had begun with photolithography, rather than metallic type, all our books would be facsimiles of manuscripts (Hodgkin 1986: 164). It is an interesting question to ask what structuring mechanisms would have been developed by the scribe to assist the reader in such an 'industrial chirographic' culture. One might predict that the prosodic use of many variations of spacing would disappear, but that use of spacing as a means to explicate structure might become formalized.

Hypertext as Structure

The following section will describe how computer based hypertexts have continued this structuration process further. The computer has been used to add structure to text, either by automating prior techniques or introducing new ones. In the same way that spreadsheets and AI are converging, so too it can be argued that recent developments in both AI and hypertext are beginning to come together, creating systems that prefigure a knowledge medium. These implementations give specific pointers as to how the design implications of the 'AI as writing' approach, as brought out in the last chapter, can be implemented.

One of the supposed benefits of hypertext is that it allows us to create texts that overcome the constraints of linearity. That is, unlike conventional written text, hypertexts can have an associative structure, each node can be linked to many others, and the user can select the path through the hypertext - the aim is thus to give more control to the user (reader).

There are a few problems with this proposal. Firstly, it is not really the case that texts as we know them are strictly linear. For example, consider a broadsheet newspaper like The Times which has achieved a presentation format that facilitates quick access (McKnight, Dillon & Richardson 1991). The use of headings and subheadings, the placing of standard material (such as the television guide or stock prices) in the same place every day, allows the average reader to get to the parts of interest very quickly. Nor has the apparent linearity of printed texts prevented the writing of non-linear material in a book format, a good example being *The Dictionary of the Khazars: A Lexicon in 100,000 Words* (Pavic 1988).

An argument against simple, arbitrary, links within hypertext has been made by both Brown (1988) and De Young (1990) who compare unstructured hypertext links with unstructured programming. "Goto's" in programming languages, by which the flow of processing control can be sent to almost any point in the code, make the reading of software code very difficult and so hinder debugging and maintenance. A similar argument has also been made against most expert systems, which consist of an unstructured set of rules, namely that is they are like 'spaghetti' programs which occasionally do not act as required (Van de Riet 1987). Of course, the situation is worse for hypertexts, as they are intended to be read all the time, whereas software code need be read only occasionally.

The arbitrary linking nature of conventional hypertexts probably has its origins in the theoretical basis of the early designs. The originator of the concept, Vannevar Bush, argued that our information technology should reflect more natural ways of organizing our material. Specifically he noted that "[t]he human mind...operates by association" (cited in Conklin 1987: 20). But writing a text in this manner is similar to the case where a writer delivers what are notes for a book as if they were the final definitive text. Associations tend to be personal, understood by the writer, and acting as a *aide mémoire.* But the reader, lacking the same context and background as the writer, requires more support to aid understanding. The associations need to be made clear. This problem has motivated the design of hypertext systems in which the links are not arbitrary but chosen from a limited set of types - structured hypertexts.

Structure

Because of the kind of problems found in the use of simple, unstructured, hypertexts, recent work has explored the value of adding structure to the loose links common in stereotypes such as HyperCard, aiding both author and reader. One approach is to constrain the arbitrariness of the links that the author can use, by having a limited number of *typed* links. Some early systems, such as NoteCards (Halasz, Moran & Trigg 1987) had typed links, but the type was specified by the author, and could be any user-chosen label. Although this aids the reader to see the relationship between the nodes that are connected, it still encourages the use of many link types.

More recently other systems, such as SEPIA (Streitz, Hanneman & Thüring 1989), gIBIS (Conklin & Begeman 1988a/b), and EUCLID (Smolensky et al. 1988), have employed a limited number of pre-defined typed links to represent the standardized (conventionalized) argument structures within the text, allowing the reader to see the rhetorical relations between the text components. Such systems continue the development that punctuation and typography began by allowing the text to be re-represented in a more structured form.

SEPIA

One good example of an active, knowledge-based authoring and idea-processing tool for creating hyperdocuments is a system called SEPIA (Structured Elicitation and Processing of Ideas for Authoring). For the representation of arguments, the SEPIA team have adopted the Toulmin Model. This model of the reasoning process considers there to be six different kinds of elements that enter into most typical arguments. First there is the *Claim*, that is, an assertion put forward publicly for general acceptance, with the implication that there are reasons that could show it to be well founded. Claims are supported by Data (or grounds) that are the particular facts about the situation. A *Datum* is connected to a Claim by a *Warrant*. These are the general ways of arguing about certain situations. The most reliable and exact warrant come from the field of science and engineering. Thus the data might be the values of known variables, and from these one makes a claim about some unknown variable. The warrant in such a case would be the relevant formula. In medical reasoning, warrants might be more heuristic. Thus if the data is a set of symptoms, such as pallor, lethargy and low fever, the claim might be a diagnosis of viral infection; and in the Toulmin schema the warrant connecting them would be a statement of the following nature: This particular combination of pallor, low fever and lethargy, which a doctor's experience enables him to recognize, generally means (i.e. entitles us to have some confidence in inferring) a virus infection. The Warrant itself can be further justified by a *Backing*, which make explicit the body of experience relied upon to establish the trustworthiness of the warrant being used in the particular case. The strength of confidence in the initial Claim is indicated by the *Modality*. Finally there is are possible *Rebuttals*, that is, the exceptional cases that might undermine the the force of the original claim.

For an example of Toulmin's technique consider the statement on a weather forecast: "The weather will be clearing and cooler by tomorrow morning". It can be modelled thus: (Toulmin, Rieke & Janik 1979: 87).

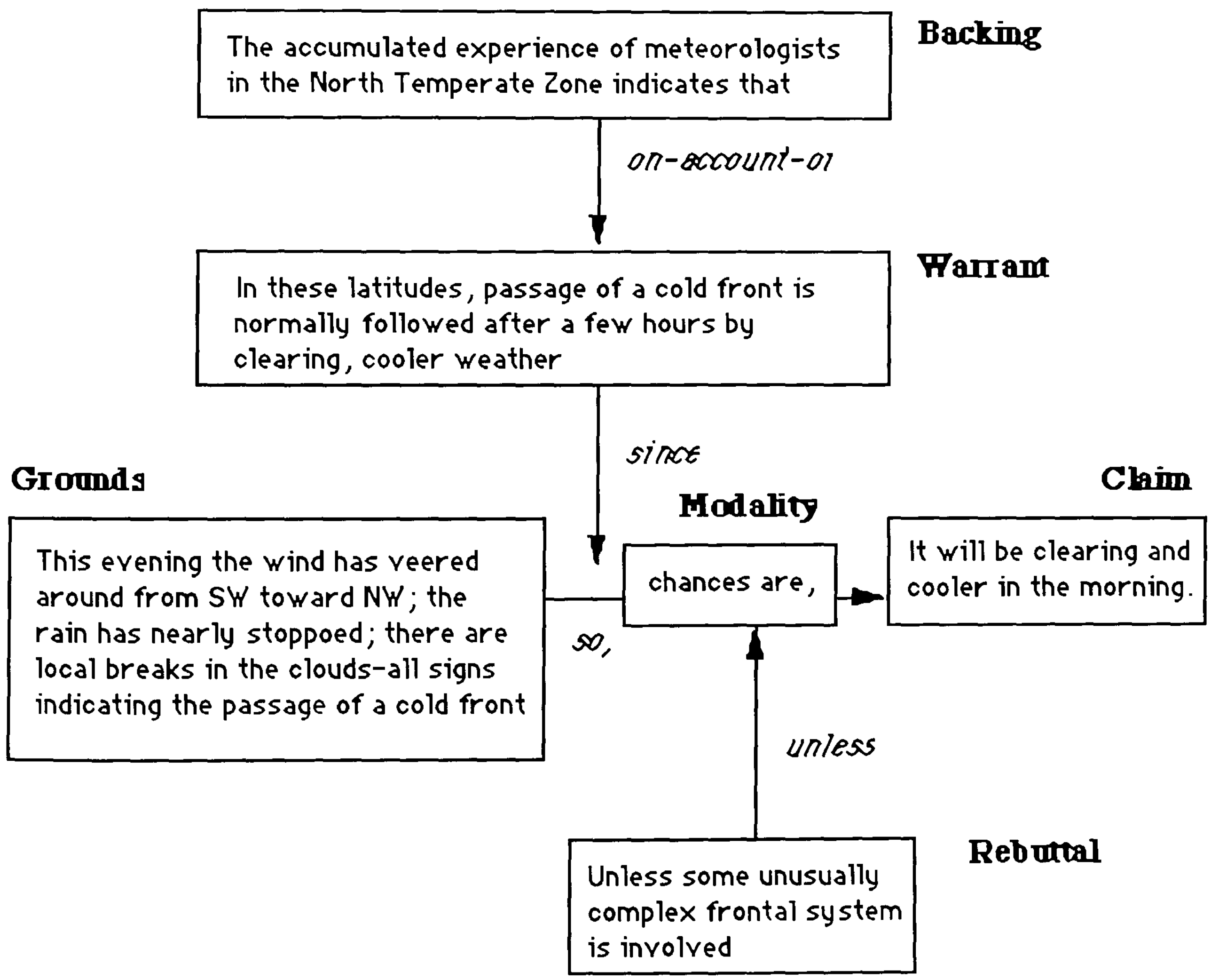


Fig. 5.2 A Toulmin Schema

The simple claim is represented with its other rhetorical elements in the structure shown, indicating the modality, grounds, warrant, backing and rebuttal. The links between the text elements are not arbitrary but have to be one of: so, unless, since and on-account-of.

The SEPIA system consists of a number of various *spaces*, which allow the writer to deal with specific writing tasks. The most important of these for the purposes of this thesis is the *argumentation space*. This "serves as a medium for generating, ordering, and relating arguments for specific issues working at one issue at a time" (Streitz, Hanneman & Thüring 1989: 12). It is this space that employs the Toulmin schema. Within SEPIA the *Datum* and *Claim* are obligatory, whereas the *Warrant*, *Backing* and *Rebuttal* are optional. The relation *so* links a datum and a claim. As the schema deals with informal reasoning, rather than formal logical reasoning, the adherence to the claim can be assisted by providing a warrant. The warrant provides the general rule which justifies the *so* link, through the *since* link. Going further back, the *Backing* provides evidence for the validity of this general rule, through the *on-account-of* link. SEPIA does not utilize the Modality element.

The nodes of the network are statements generated by the author, each of which has four attributes: position, claim, datum, and level. The values of the first three are set to either true or false, according to the author's subjective view of the nature of the statement: that is, is the statement a position, a claim or a datum? The level attribute is a measure of the level at which the node stands. This is important because in SEPIA a statement can be a datum at one level and a claim at another, the data can be brought into question. Thus SEPIA allows hierarchies of links going down through a series of interconnected levels. The system facilitates the construction of arguments by providing templates of the rhetorical elements required by the author. Thus the author can use the SEPIA *object-to* operation to attack a claim already in the system, which can be further refined by the operation *justify*. This will return a new window with a Toulmin schema structure template for the author to fill in the required warrant and backing.

The Toulmin Model is currently popular within the hypertext community. It is also being used, for example, by Marshall and Irish (1989) within NoteCards (a hypertext system) to represent legal argumentation (see also Newman & Marshall 1992). The Toulmin schema was developed to be able to represent as wide a set of reasoning types possible, Toulmin, Rieke and Janik

(1979) show how it can be applied to specific arguments in the law, science, the arts, management and ethics. There are other structuring schemas, however. One of these has been specifically designed to represent design issues, and is implemented within a hypertext system known as gIBIS.

gIBIS

The gIBIS (Conklin & Begeman 1988a/b) system implements a paper-based methodology known as 'Issue Based Information Systems' (IBIS) which was developed for use on large and complex design problems. The IBIS model is similar to the SEPIA implementation of the Toulmin schema in that it provides a limited set of node and link types representing a theory of how argument and reasoning proceed. IBIS focuses on key *Issues* in the design problem, each of which can have many *Positions*. A Position is a statement or assertion which resolves the Issue. Each Position may have one or more *Arguments* which either support, or object to, it. These are placed in hypertext nodes.

There are seven types of link within gIBIS that connect these nodes.

Position	*Responds-to*	Issue
Argument	*Supports*	Position
Argument	*Objects-to*	Position
Issue1	*Generalize*	Issue2
Issue1	*Specialize*	Issue2
Issue	*Question*	Issue, Position or Argument
Issue	*Be-suggested-by*	Issue, Position or Argument

Table 5.1 Link Types in gIBIS

Two other special link types are added to the original IBIS model:

Other allows for users who cannot express a link using the above links.

External for linking to non-gIBIS material such as sketches.

The author can create a hypertext document by putting appropriate texts into nodes and connecting them with the typed links as required. The reader can then browse the network seeing how the argument is constructed. The interface is that of the traditional hypertext system: a map has icons for each node and the reader can select any node to read its contents. When the system is being used in a collaborative mode, the reader can also become a co-author, adding new nodes to develop the argument.

Another modification of the IBIS approach, called PHI (Procedural Hierarchy of Issues) has also been implemented. One application of this methodology, the *Author's Argumentation Assistant* (AAA), has combined the IBIS/PHI argumentation model at the macro level, with the Toulmin structure at the micro level (Schuler & Smith 1990). This combination is achieved if the Toulmin 'claim' is identified with the PHI 'position', and the Toulmin 'datum' with the PHI 'argument'. The bringing together of the various argumentation rhetorics is an issue that arises later, when, as we shall see, these same claim and datum can be identified further with the rule-base system employed by many AI systems (an integration hinted at by the close similarity of the medical Toulmin warrant example above and the medical expert systems discussed in chapter 2).

EUCLID

Smolensky and his colleagues (1988) have created the specification for a tool, called EUCLID, which helps people create and assess arguments, and communicate these to others. It provides users with a language, ARL, for expressing their arguments in a clear and standardized way. They, too, aim to use the computer as a medium, not as a replacement, for human reasoning.

The approach, like that of SEPIA and gIBIS, takes a line which will appear in the knowledge medium described in this thesis, in that there is a separation between formal and informal expression. The system has access to and manages formal descriptions of arguments, but the language content of these is informal. EUCLID is intended to provide a unified environment for working out an argument and expressing it in text. It is a specially tailored hypertext system for reasoned discourse. The nodes are similar to those in SEPIA and gIBIS being: terms, definitions, claims, arguments, documents, strengths, and domains of assertion.

Problems of structured Hypertexts

Although an improvement on unstructured and arbitrary hypertexts, the structured hypertexts have proved not to be without problems of their own. One of the problems that was uncovered in the use of gIBIS was that of *context* in non-linear documents. Readers found they could not follow the author's line of thought (they lost the thread of the argument). The designers concluded:

> we suspect that there is a ... more subtle issue here: that traditional linear text provides a continuous, unwinding thread of context as ideas are proposed and discussed - a context which the writer is directly, if unconsciously, constructing to guide the reader on the salient points and away from irrelevant and distracting ones. Indeed, a good writer anticipates the questions and confusions that the reader may encounter, and carefully crafts the text to prevent these problems (Conklin & Begeman 1988).

The problem is that the author cannot anticipate all the possible ways in which a reader may arrive at a node in a network. Indeed, for certain literary uses of hypertext this is exploited. In Michael Joyce's *Afternoon*, many nodes use pronouns, and the specific instance of who is doing what depends on the readers path to that node, as the readers notions of who 'he' and 'she' are will vary. However, for general argument construction, this ambiguity appears to be a problem, and the knowledge medium, as mentioned above, dispenses with the notion of a purely non-linear text.

One implementation of the IBIS methodology is designed for simple glass-teletype terminals. It is called itIBIS (indented IBIS), as employs indentation of straight text to represent to hierarchical relationships between the nodes (c.f. Sowa). Letters are used to mark the type of node that is listed: I for issues, P are positions, AS for supporting arguments and AO for objecting arguments. The resolution of issues is marked with *, indicating that a decision has been made. If no decision has been made a '?' is used, and '-' shows rejected positions. Initial results of its use in the client organization showed that those in the team using itIBIS found it easy to learn and very useful in capturing design rationale (Yakemovic & Conklin 1989). However, later they found that there was a problem making the results of itIBIS available for those outside the team using the system regularly, it took some experience with the format to be able to understand it. This difficulty was addressed by converting the output into a more conventional prose form (Conklin & Yakemovic 1991: 379):

itIBIS form:
I: How should we handle situation X?
    *P: Using Y
        AS: simple
        AS: some of this work needs to be done anyway
        AS: may be able to borrow from an existing implementation

    -P: Using Z
        AO: <etc>

Prose Form:
The first issue we addressed was how to handle situation X. The first solution we discussed, and the one we finally selected was using Y. This approach has the advantage of being simple. Also some of this work needs to be done anyway. In addition, we may be able to borrow from an existing implementation. Another potential solution discussed was using Z, which was rejected because <etc.>

Fig. 5.3 Standard and Prose Forms of itIBIS

One of the most interesting features about the development of EUCLID is that in early proposals (Smolensky et al: 122-123) the system is illustrated by relating the structural analysis of the argument *next to* and being *explained* by the full text of an argument. Later (Bernstein n.d.: 8), the system as implemented adopts the approach seen in SEPIA and gIBIS, where the text of the argument is placed in small windows that are linked together — although other work (Bernstein, Smolensky & Bell 1989) structured the graphical display of the windows to represent the structure of the underlying arguments. The knowledge medium approach, however, takes the view that as writing was supported by speech in the early developmental stages, so a formal language of argument representation can be best served by having full, conventionally written, text to support the reader's interpretation of the formal structure. One convention that this text provides is the linearity, the "continuous, unwinding thread of context as ideas are proposed and discussed" (Conklin &Begeman 1988a: 327), that ItIBIS users seemed to miss.

Integrating Structured & Computational Hypertexts

The most interesting developments, in so far as the theme of this thesis is concerned, are those systems, currently being designed, in which elements of both structured and of computational hypertexts are integrated. For example, the notions of structured hypertext, such as SEPIA, have been made executable (Clark 1991). Clark has interpreted the components of the Toulmin model into elements of a conventional rule-based system, as used in most expert systems:

Datum = Facts

Claim	=	Conclusion
Warrants	=	Rules
Modifier	=	Degree of certainty

A rebuttal R of a rule if A then B can be included in this format with a rule in the form

if A and not R then B

The important step in Clark's work is to recognise that the warrants of the Toulmin model are similar to rules in rule-based systems; both encode general relations between specific data and conclusions from that data. Given this approach the various Toulmin structures that underlie an argument can be executed on a computer using a rule-based system. Clark's main intention with this approach is to facilitate reasoning in areas of disagreement. This is because the benefit of using a rule based system to implement warrants is dependent upon the generalizability of each warrant/rule. Clearly, if each warrant is only executed once, the full power of rule-based systems is reduced.

Clark's system, called 'Optimist', as a rule-based system, has a set of rules. The user can modify the strength of these, and add a new (and purely textual) backing to justify this change. That is, multiple use of a rule is made by having the strength of the qualifier vary according to various users' beliefs. A new user can explore an argument and vary the strengths of qualifiers, if he or she disagrees with the original author, creating his or her own set of rules (copies of the base set). If necessary a setting of zero can be established to disable that particular rule (i.e. if one believes that the rule is totally irrelevant). The rule-base is then executed and creates a new argument. The users cannot edit the basic rule-base, only the degrees of certainty. The user then adds backing to the argument chain by associating typed, informal, text to the warrants. It is this aspect of Optimist that relates it to structured hypertexts. The backings are non-executable explanations of the author's rationale for selecting the degree of certainty chosen.

In Optimist, sample warrants (i.e. rules) are collected together in a particular 'model', typical warrants being (Clark 1991: 65):

```
skelwarrant 1 ::
if        'reservoir mapping' of prospect is direct
then      trap is effective
```

```
skelwarrant 2 ::
if          'reservoir mapping' of prospect is 'indirect, constant'
then        trap is effective

skelwarrant 3 ::
if          'reservoir mapping' of prospect is 'indirect, anomalous'
then        trap is effective

skelwarrant 4 ::
if          trap of prospect is faulted
and         'most recent faulting' of prospect is Rock1
and         'seal unit' of prospect is Rock2
and         Rock1 is_younger_than Rock2
then        trap is effective
```

Table 5.2 Examples of Optimist skeleton warrants

This model also has strengths and backings associated with these skeleton warrants, giving them specific values for the probabilistic mechanism:

Warrant	Strength	Backing
1	100	'Direct mappings perfect by definition'
2	80	'Constant corr. proven reliable in 14th round'
3	40	'Anomalies serious here (see p43/21-1)'
4	100	'Late throws no problem as calcite conc. high'

Table 5.3 Strengths and backings in an Optimist model

A user working with the system can, if they disagree with the line of reasoning, edit the strengths (by direct manipulation of graphical 'sliders'), and add his or her rationale for making changes. The set of changes creates a new model: a new collection of strengths and backing for the same skeleton warrants.

Stutt (1989) has also implemented Toulmin structures, but his approach is to take the machine as a critic of the user's arguments, whereas Clark takes the approach advocated in this thesis of using the system as a medium between human co-workers.

The relation between the executable and non-executable parts of these systems is crucial, and the way these point to a close integration of the two types is one that Conklin recognized. There is a close similarity between hypertexts and the semantic networks employed in AI (Churcher 1989), where a semantic network is a directed graph connecting concepts in which the links represent the various relationships between the concepts:

> The difference is that AI knowledge engineers are usually striving to build representations which can be mechanically interpreted, whereas the goal of the hypertext writer is often to capture an interwoven collection of ideas without regard to their machine interpretability (Conklin 1987: 37).

Because structured hypertexts deal with the way that an author or reader thinks about the material within a text, and computational versions of these Toulmin structures are programs that represent modes of thought, the link between AI and hypertext can been made. Within the AI community there has been a call for such a merger to be made, although few systems exist (but see, Rada 1990; Carlson & Ram 1990). This is probably due to resistance from the AI community, which still sees its role as the creation of dialogue. It is also noteworthy that there are, in contrast, attempts to impose a conversational mode of interaction to hypertexts (Patrick & Whalen 1992).

The situation is different, however, for those AI workers who are involved in the analysis of human performance to inform the design of AI systems. As the user of this work is a knowledge engineer, and not a member of the public with little or no computer literacy, their immediate concern is not with a conversational interface, but with analysing the material gathered. Analysing the source material - known as knowledge elicitation - conventionally means dealing with either transcripts from interviews with experts talking through a problem, or published books in the domain. The sources are, in other words, texts. As Regoczei & Plantinga note "the analyst is after knowledge, but all he gets are words" (in Motta, Rajan & Eisenstadt 1990: 22). For this reason, those engaged in creating tools to assist in knowledge elicitation have created systems that marry aspects of text manipulation (hypertext) and a cognitive model (knowledge-based system).

For the past few years research at the Open University, in particular, has been investigating tools to assist the knowledge engineer. The computational analysis of parts of a text, in this case a transcript of expert behaviour, is made and executed. However, the implementations arising from this task are, at the end of the day, not bound to the original source text. They are meant to run alone in the standard question-answer mode characteristic of expert systems. The Running Text program can be seen as keeping the model and original text together, the AI program component being an interpretation of the text, one that a user could modify and re-run; alternatively, one can see the text as a means to explain to the user material that is difficult to formalize.

An early system that the Open University group implemented, CREF, was a special form of text editor that allowed the knowledge engineer to break up the transcript or document into discrete units and to connect these into chains of argument. CREF could be seen to support the task of the knowledge engineer

> with facilities that can be thought of metaphorically as a means of weaving arbitrary coloured threads through pieces of text (segments) and then allowing the user to pull on the threads at whim in order to obtain selective views (collections) of the transcript (Motta, Rajan & Eisenstadt 1990: 33).

Acquist is a hypertext-based component of a larger knowledge engineer's workbench, designed as an improvement on CREF, which "leads to the construction of an abstract architecture of the domain that is *tied-to* and *supported-by* the fragments of text from the transcript" (Motta, Rajan & Eisenstadt 1990: 35). CREF only tied together elements of text, Aquist facilitates a more detailed description of the rhetorical relations between the text fragments. Acquist has a set of domain independent links that can be used to connect items in the transcript. The following are provided (Motta, Rajan & Eisenstadt 1990: 38):

•NOTES links an item to a note, for the knowledge engineer's own arbitrary reasons.

•SPECIAL-CASE links one fragment to another for which it is a special-case.

•JUSTIFIES links a statement to another that provides a justification for it.

•EXPLAINS links a piece of text to a fragment that clarifies or gives a rationale for it.

•REINFORCES links an assertion to a fragment that has information strengthening that assertion.

•WEAKENS links an assertion to a fragment that has information weakening that assertion.

•INCONSISTENT-WITH links two mutually inconsistent fragments

•DOCUMENTATION links fragments to be used to document a concept.

These links are very similar to those offered in conventional structured hypertexts such as gIBIS, SEPIA and EUCLID. Their role differs, however. They aim not assist a general reader, but to assist the knowledge engineer in creating computational models of the expert behaviour within the transcript. A similar knowledge engineering support tool, CONCORDE, has also been described in the literature (Hofmann, Schreiweis & Langdörfer 1990).

One system that is being designed to integrate both expert systems and hypertext technology at the end-user, rather than knowledge engineer, level is IMAD (Hayes & Pepper 1989). IMAD behaves as a "smart book"; it integrates a hypertext version of a manual for a piece of equipment, and a diagnostic expert system to help solve faults. The hypertext component is added

> because of practical limitations to current diagnostic technology [an expert system] ... [which] does not allow a technician to browse for greater understanding or insight into the system being maintained (Hayes & Pepper 1989: 126).

The textual component within IMAD overcomes this problem by providing additional context material to aid greater understanding of the diagnostic mechanism. Like IMAD, Frisse and Cousins (1989) have brought together aspects of hypertext and reasoning techniques, although they come from a concern with improving the ease of use of hypertexts, rather than of expert systems. They have implemented special indexes for the 'Dynamic Medical Handbook Project which can be interpreted as belief networks, and hence exploited for automated reasoning.

Both traditional AI and traditional hypertext systems have suffered from a similar problem. They decontextualize knowledge, taking it away from an original linear and bounded 'text'. A knowledge-based technology such as IMAD is addressing this problem through the use of text, but often without recognizing that hypertext workers themselves can identify a similar problem with traditional hypertexts. An important component of the information conveyed by an author to a reader in a traditional setting is the order in which the material appears. In most current hypertext systems, readers may fail to understand the material presented because they view it in the wrong order, or they may simply comprehend it less well. (Zellweger 1989: 1). The same point has been made by Marshall and Irish:

> Although methods for maintaining coherence are more or less settled for conventional forms of writing, hypertext violates many of the assumptions underlying these methods ... One important aspect of

> conventional forms absent from hypertext is the transitional text that helps the reader maintain a sense of the material's coherence. The fragmentation characteristic of hypertext may also lead to a lack of interpretative context (1989: 22).

To address this problem hypertext researchers have begun to re-introduce a form of linearity. For example, Zellweger (1989) employs the notion of a *path*, an ordered trail through the material which the author sets up to aid the reader. Marshall and Irish (1989) call their trails *Guided Tours*. This idea actually goes back to the beginnings of hypertext, when Vannevar Bush argued that his system should have trails; each trail acting essentially as a new 'book'.

In the same way that authored paths have been introduced into hypertext databases — so the knowledge medium introduces authored, directed, paths into knowledge-based systems. The system is no longer only a knowledge-base that can be queried, it is given an authorial presence - a series of points that address a concern imposed by the author. This is an important design feature for the knowledge medium. It will use a series of question/answer pairs derived from knowledge bases, but these are then strung together into a (basically) linear argument. Of course, one does not lose all the conventional features: the reader can query the knowledge bases and ask for new answers, exploring the implications of the authored text, possibly using these answers to write a new text.

In this chapter it has been argued that developments in hypertext are leading to a convergence with AI technology. The need to impose a structure and the desire to exploit the computational power of the machine have lead to the design of hypertexts that compute argument structures. The possibility of the marriage of the two has been particularly exploited by those engaged in knowledge elicitation, to aid AI researchers who have to deal with texts. Any new system should also apply the recent lessons of hypertext use and employ paths to provide an interpretatively coherent structure for the reader. It is this lesson that the knowledge medium proposed in this thesis will apply - the reader initially sees an essentially linear document.

The last three chapters have used the history of writing and of hypertext to inform design decisions for any use of knowledge-based systems. We are now in a position to put all these together. The following chapter will describe a specific implementation of a knowledge medium — called *Running Texts* —

a prototype system that this thesis proposes as a close integration of AI and hypertext systems, and which applies the points that have arisen in the previous chapters.

6. Running Texts

The themes that have arisen in the previous chapters can be used to inform the design of a new system that adopts the 'AI as writing' metaphor, and the notion that symbolic AI is converging with hypertext (there is also research that integrates hypertext with connectionist approaches to AI, see Koch 1992, for example). The parallels have been of a broad nature, and the design guidelines from them would allow a multiplicity of potential implementations for a knowledge medium. This chapter concerns itself with a more specific design. The specific implementation details are not arbitrary, but are motivated by additional parallels with the domain of writing: Issues such as the role of questions in rhetoric, the idea of explanation as decontextualization, and the layout of the glossed Bibles of medieval Europe all contribute to the final proposed design. The particular implementation, and the relevant issues that have assisted in its design are described.

The specific system being described below is known as the *Running Text* system. It is so called for two reasons:
1. Unlike conventional AI system, the user initially deals with continuous, *running*, text, not only the short questions and answers common in knowledge-based systems (although questions and answers are there, initially hidden from the reader's view).
2. Unlike printed texts, the AI components that underlie the normal text of running Texts can be executed by the computer, that is, they can be *run*.

The use of the term running taken from both the domain of written text and from computing denotes the aim of the work to contribute to the integration of the two technologies.

Summary of Design Decisions

From the discussions in the previous chapters the following broad design criteria for the Running Texts have been established, and are summarized below:
- We can use discrete and separate patterns of reasoning (cf. interpretation models), in the examples given they are a rule-based and a causal reasoner (Fig. 6.1 a).
- Due to the nature of AI as a technology based on questions and answers, the latter can be the basis of the units that an author assembles to create the final

'text' (Fig. 6.1 b), with additional reader access to the questions.
- The need to assist in conventionalisation suggests a conventional text which is visible in parallel with the formal expressions; the conventional text also allows one to represent material that is currently difficult to implement formally (Fig. 6.1 c).

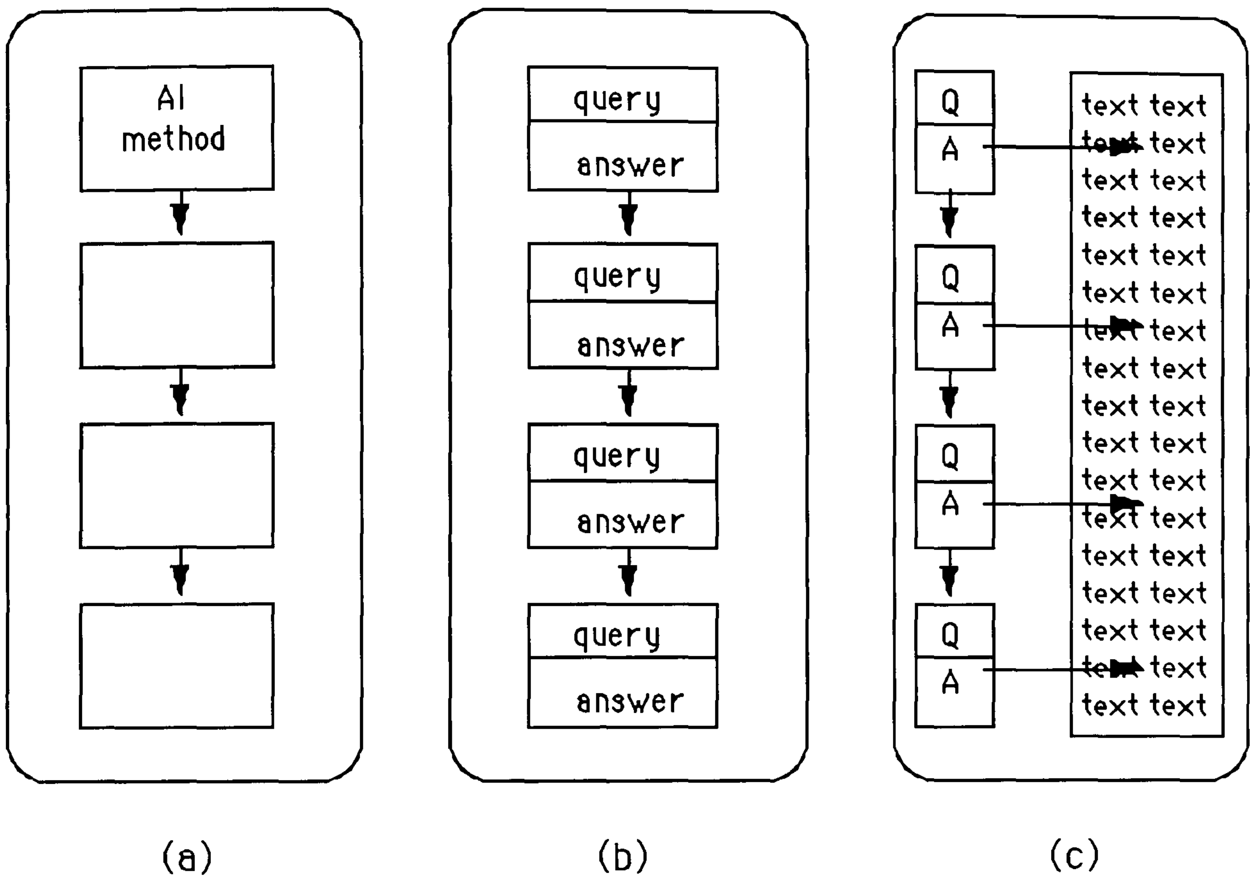


Fig. 6.1 Overall Design for a Knowledge Medium

There is an additional complication in that the underlying model needs to be visible to the user too. The 'model' is the knowledge-base over which the query applies, and from which the answer is generated. An initial design tried to place all these elements together on the screen for the reader to see. That is, at any one moment, the text for the query, the basic model, the explanation generated, and the free text to which is connected, were all on screen. But due to the sheer volume of text, this solution proved unworkable for any real application. The need to compromise between the need to display all relevant text for the reader, and the requirement to keep the screen size to an acceptable, is one that constantly arises in the design.

So the decision was made to separate out the 'query/model/answer' (the Q/A in the figure above) component from the 'answer/text component'. The answer that is shared between the two allows their relationship to be made clear. Each query/model/answer is created, viewed and edited in its own editor (a multi-pane window), known as a *Thread Editor*. It is so called because each answer is known as a *Thread*, which itself is so named because many such answers could be (metaphorically) tied together within the final text. The answer/text component, that which the reader is initially presented with, is created, viewed and edited in another multi-pane window, known as a *Running Text Editor*.

The Prototype Development System

The example Running Text system described below is implemented in Smalltalk/V, an object-oriented language, running on the Macintosh computer. Smalltalk/V was selected for a number of reasons. Firstly, it has objects already available for such interface features as windows and menus. The use of these graphical user interface mechanism is important to ensure that the reader sees the system as a writing aid, the aim is to avoid as far as possible implying any sense of dialogue.

Secondly, the Smalltalk/V environment exhibits some of the developments described in chapter 4. The programmer is presented with workspaces, which are text panes. These panes are like those found in a text-processor in that they have simple text editing functions, but they also offer additional features. The programmer can type in Smalltalk code and, once selected with the cursor, this code can be executed. This proves very useful for testing small bits of code. This ability is also used to provide tutorial documents which describe the Smalltalk programming language, and give simple examples that illustrate various programming constructs. This feature is exploited by the Running Text, which allow certain of the text panes to contain text that can also function as executable code.

In fact, this ability of Smalltalk/V and its multi-pane window feature makes it very appropriate to undertake the integration of dynamic and structured hypertexts as recommended in Chapter 5. One way of seeing the Running Text is as the use of Smalltalk/V's pane features as a means to restrict the types of information (to be a query, a rule-base et.) that occur within a pane, and to regulate the connections between these panes types. Some of the panes are computable, others not.

The Inference Engine

Most threads in the examples below employ a forward chaining system (a form of rule-based system) as the reasoning mechanism. This starts with known facts and applies the rules to them, generating new (inferred) facts. Of all AI techniques the rule-based system (Hayes-Roth 1985) was selected because it is both the most common form of inferencing mechanism used, and it is also one of the most general.

The example to be described below has been implemented with an existing inference engine, available from the Smalltalk/V supplier (Digitalk 1991). The decision to do so, rather than implementing an inference engine anew, was made for a specific reason. Because the ideas in this thesis are based on the reader being able to re-use the materials from a text that has been read, it seems appropriate to begin this approach by using an existing tool. If it is found insufficient for the task, this would provide evidence that the notion of re-use is questionable. If the author and readers had to create new code for every thread, the system would offer little for the user.

Thread Editor

The Thread Editor is a three pane window. The query appears in the upper left pane, the model into the lower left pane, and the answer that is generated by applying this query over the model is in the larger right hand pane. In many cases, the model that the author employs will be the same for many threads, that is, he or she will simply apply different queries over the same model. Clearly it would be inefficient to have multiple copies of large models existing in each thread. Thus, there is also an option to save a model in a general model library, and equally, to set the model for a new thread to be an existing model from this library of models. A similar approach has been employed by Huuskonen (1992) who has also separately modeled questions and answers associated with the tasks that the system undertakes.

Queries and Questions

One pane contains the query. This is a formal representation of the question which is to be put to the model. For example, in the first example below the basic query is:

```
JCT 87 - is query
```

which can be interpreted as a request to prove that 'JCT 87', that is the contract type JCT 87, is the appropriate choice in the circumstances. These particular circumstances are also included as part of the query, as a set of facts. This is necessary to reduce the size of the final system. If the model contained the specific facts of each situation, then each thread would have a unique model differing in only these facts, all the rules would be the same. A tremendous overhead. By having the facts as part of the query, the model can consist of the general rules. In terms of the Toulmin schema, the query pane contains both the claim and the data, while the model pane contains the warrants. The term model is more appropriate for the causal thread that will also be described. In the implementation the author can input the facts and query through a simple menu which contains a list of those available in the model.

The query (or question) is a fundamental part of the proposed design. This was initially motivated by the recognition that AI in its history (through the Turing Test) and its techniques, has a fundamental affinity with the question and answer pair (Of course, this point does not relate to other areas of AI such as vision systems and robotics). However, this aspect of the design finds echoes in the philosophy of text developed recently by Meyer (1983). His position is that language use involves the resolution of a question. The answer can be one term, one sentence, or a whole text. Meyer argues, in fact, that "Questions and answers are the units of thought, not statements or signs" (1983: 4). The role of questions runs deep in his theory of language. He believes that man and women act in terms of the problems that life throws at them, and language contributes to the resolution of these problems. Thus, with language, one either expresses the problem because its resolution depends upon another, or one passes the solution onto another. Meyer goes further, and argues that even apparent statements are best seen as the answers to unstated questions. What is crucial in argumentation is "the process of making answers problematic" (Jamison 1991: 66). If Meyer is correct, then framing the material to be 'written' as a series of questions and answers should prove possible.

This question-answer pair has also been shown to be useful in practical system design. For example, questions are a crucial part of the computer-based argumentation system PHI (Schuler & Smith 1990), mentioned in chapter 5. Debate within PHI is represented as being a set of 'issues', 'positions' and 'arguments'. What is significant about PHI is that it rigourously insists that the 'issues' are expressed in the form of questions, to which the 'positions' are answers. The 'arguments' either support or refute 'positions'. Similarly, in the area of writing conventional technical manuals, Jaynes, in arguing against the unstructured hypertexts discussed in chapter 5, suggests that readers need structure. One structuring mechanism he proposes is the provision of "meaningful answers to specific problems" (1989: 159).

The Model

The Model in most of the instances below is what traditional AI would term the knowledge-base. In the architectural contract example, worked through in this chapter, it consists of the set of rules. An example of a rule from this particular model is:

```
number: 5
condition: [
#(job isSmall) isFact
& #(between 15,000 and 100,000) isFact]
action: [#(MW 80 or ASI SW)]
description: 'use contracts MW 80 or ASI SW'.
```

This rule can be interpreted as meaning the following: If the architectural job is small, and the cost is between £15,000 and £100,000, then use contract types MW80 or ASI SW. The particular formalism used in this example is a very simple rule system, only allowing logical conjunctions. It is not possible, for example, with the inference engine provided to formalize conditions that are disjunctive. One feature of the rule system employed is that each rule can be assigned a free text 'description' for the action(conclusion) part. This allows the author to write a more readable version. It is this text that is used in the generation of the answer.

The Answer

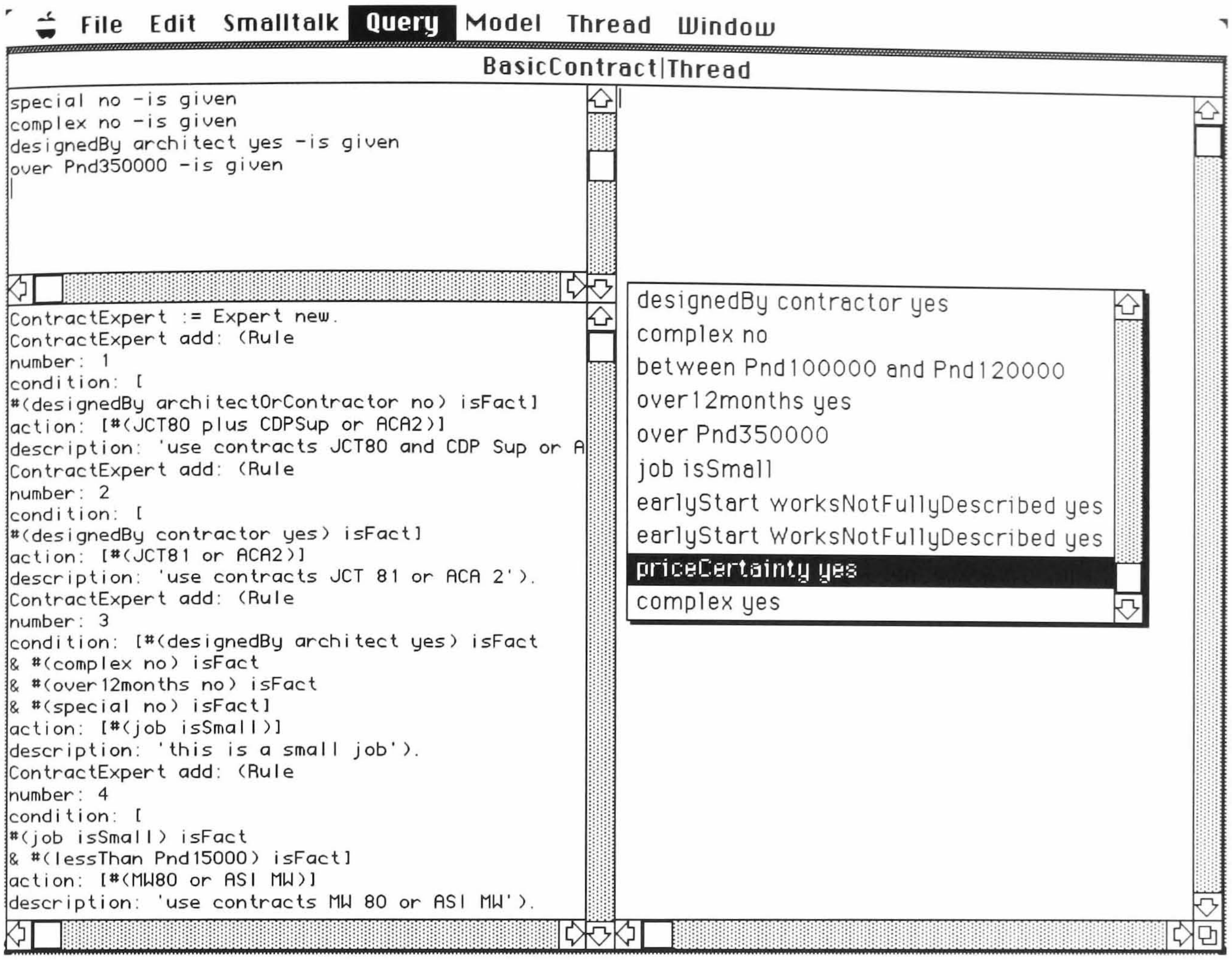
The query, when applied over the model, generates an 'Answer'. The text of the answer appears in the right hand pane, and it is this text that is passed over to the TextEditor when required. The answer is not just a simple answer, but for reasons outlined in chapter 2, the answer is a trace of the reasoning. The role of this answer as an expert system explanation is not without consequence. Given the traditional 'AI as agent' metaphor, an explanation would be designed to be a simulation of the explanatory behaviour of a human given the same query. But what is the rationale if we follow the 'AI as writing' approach? What feature of writing rather than human performance are we implementing. In fact there is an aspect of writing that matches closely the role that explanation fulfills. When one writes it is usually the case that one is not quite sure when or where it will be read. Writing requires an explicitness "if meaning is to remain constant beyond the context in which is was originally written" (Applebee 1984: 577). This theme arose in previous work on explanation undertaken for the DecisionLab project (Huxor 1988). It became clear that a good explanation allowed the listener to apply the lessons of the utterance by separating an answer from the immediate situation. So that, for example, an explanation of the effect of a rise in taxation on employment will not only deal with the specific time and place, but deal with potential effects in other places at other times. An explanatory answer becomes de-situated, which is also a requirement for good writing, as writing achieves its full potential when its permanent nature allows it to have meaning elsewhere. Through explanation, the reader not only gets an answer but can see the connections between the various components, and those things that are given, givens that may change. Indeed, it is possibly not too radical to propose that the current notion of what explanation is owes much to the effects of writing being internalized by literate cultures, and affecting our spoken forms.

Creating a Thread

To create a Thread the author calls up an empty Thread Editor, and creates a new named thread — each thread has a name, by which it is constantly referred. He or she then inputs the model and the appropriate query. In the example below, the thread is based on a rule-model. To assist in the construction of the thread, a simple interface has been added to this thread type which allows the author to select the facts that are relevant, and the goal that the system must prove, through a simple menu selection.

To summarize, the case illustrated does the following. Facts are selected from those available to fit the known situation. For example, '*over 350,000* ' represents the fact that the building project in question has a cost of over £350,000. Other facts, on the known price certainty and early completion of the job are also chosen. After selection the system places these facts into the Query Pane (see Figure 6.2). Once all the known facts have been input, the author selects the goal to proven, also through the same menu, and this goal is placed in the query pane as *JCT 87 - is query*. The same command also asks the system to take the facts above and show that, given these, it is true or false that the contract form 'JCT 87' should be used for the job. If this conclusion can be proven, an explanation is generated which lists the inferential steps. It is this explanation that is placed in the Answer Pane, on the right-hand side (see Figure 6.3). If the goal cannot be proven given the facts available, the author is informed through a message box.

The answer generated has two features which lead to the selection of the inference engine for this implementation. As it is intended that readers would see this answer, the ability to have a slightly more 'English' version of the formal representation aids user acceptability. Of course if readers then desire to edit the thread for their own use they must also understand the deeper (formal) representation, but it is imagined that there would be differing levels of user use. As they gain confidence in AI representations, they would go deeper into the 'text'. Secondly, reading the answer generated is further aided by the use of indenting each step to indicate the structure of the underlying rule matching. Again, this means that a novice 'reader' gets a better idea of the structure of the inferencing, and not just the individual steps, without having to view the model itself.



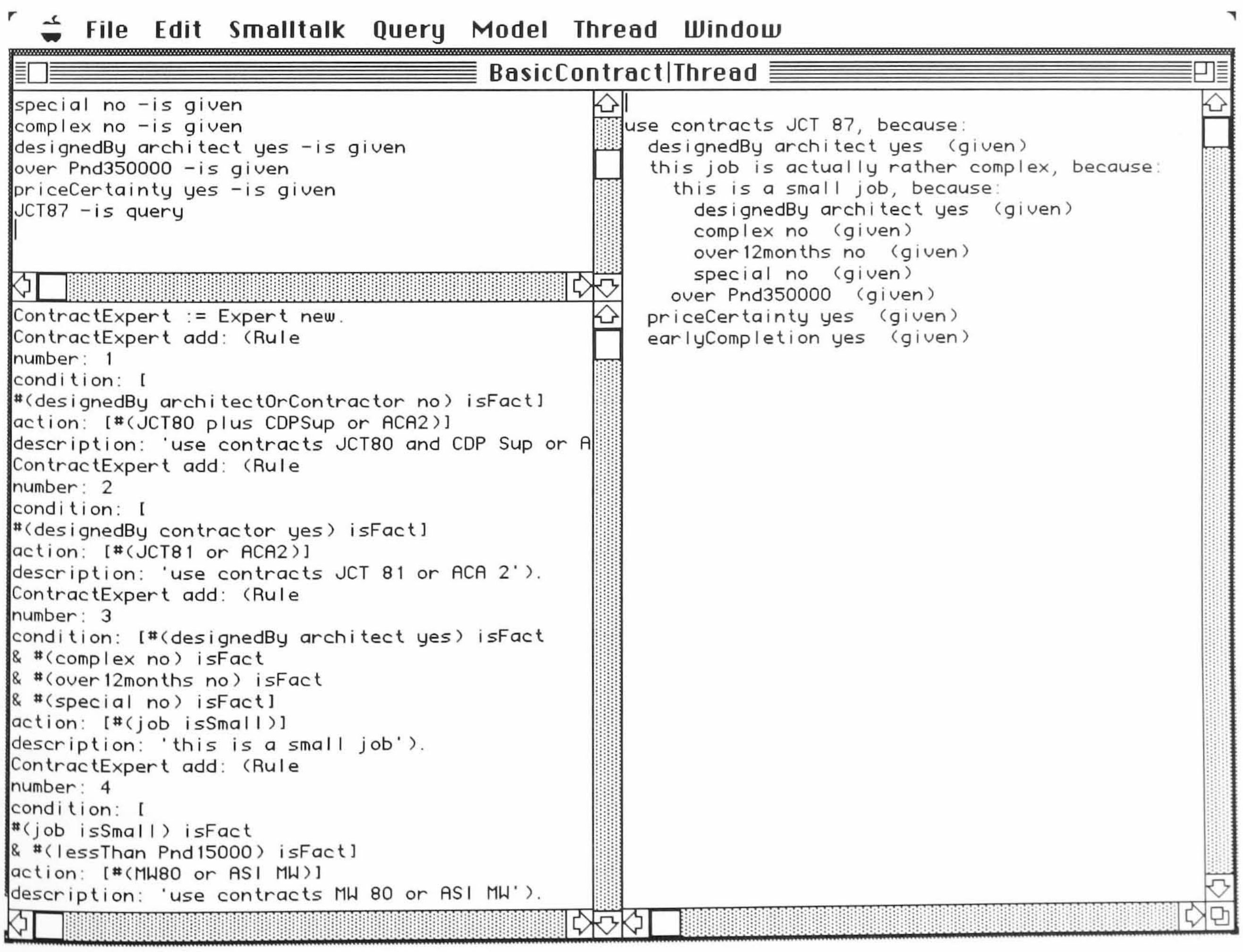


Figure 6.2 Thread for Architectural Contract Selection

TextEditor

Once the author has created a number of such threads, these are assembled to form a text, although of course only one thread may be necessary in certain cases, such as that being worked through here. The coordination of the various threads, and their linkage to more conventional text takes place in another window, the *TextEditor*. Although called an editor, this window is also the Running Text viewer — the reader gets to read the text through this window. The TextEditor consists of two panes:
1) the right-hand *ConTextPane*, which contains text known as the *ConText*.
2) the left-hand *GlossPane*, containing another text, known as the *Gloss*.
The GlossPane contains the various answers from the threads that will constitute a Running Text, and their associated conventional text is found in the ConTextPane. When the answers have been passed over into the Text Editor from a Thread Editor, they are known as 'glosses' rather than answers, for reasons that will become clear. The essential role of the TextEditor is to allow the author to call up various answers from the appropriate threads, write the associated free text, and create links between the two.

Typographical Layout of the Text Editor

The Text Editor, then, has two components, free written text and a set of computationally derived glosses. One design issue is how these are to be coordinated: in other words, how is the text to be laid out to ensure that the relation between the conventional text and its associated glosses is made clear. The literature review has suggested two schemes. The first is based on Biblical Glosses (de Hamel 1984), in which the gloss text is placed next to the text to be explained. The second based on the more recent Mathematica NoteBook. A Mathematica display is illustrated in Figure 4.2, and consists of a set of different paragraph types, one after the other. When applied to the Running Text this approach would give alternative sections, in linear order, of free text and computationally derived text. The example implementation is based on the gloss design. It would, of course, also be possible to create an interface which combined both. The Biblical glosses are more complex, have had a long history, and deserve a more detailed explanation.

Fig. 6.3 Prince Henry's Canonical Epistles glossed
(de Hamel 1984: plate 4)

The glossed Bible in its many variants shares many features with the requirements of the Running Texts. In the glossed Bibles, the scribe copied two texts: the biblical text and the Gloss which explains it, both of which were written in full. The gloss either next to, or within, the Bible text (see figure 6.4)

The gloss was placed as close as possible to the biblical text that it was explaining, as it was important that no confusion should exist in the reader's mind about the connection. A Bible reader could note explanations, to assist in understanding the Bible text, without having to lift his or her eyes from the page, and without the need to get another heavy text down from the shelves. Equally, a reader knowledgeable of the Bible text could use it to access the comments in the gloss. Another feature that the Bibles share with our domain is the concern over the utility of space. In the case of the Bibles, parchment was expensive and over time the scribes developed various methods of reducing the amount of empty space. Similarly, the Running Texts, if they are to be available on small tablet style computers, need to reduce the screen space required while still making the text clearly legible, where the text includes both conventional text and its relevant queries, models and answers.

The process of increasing the space efficiency of gloss and Biblical text coordination took some time. At first the scribes took the most obvious solution. The complete Bible text was written in a central column. They then went back to the beginning and wrote the appropriate glosses: short glosses between the lines of the Bible text, longer glosses in the columns to the left and right of its associated Bible text. This gave the traditional gloss/text/gloss layout. Although rather inefficient in use of space, this method did allow readers to add their own glosses in the many blank spaces that remained. This tradition of allowing readers to appropriate text is one that many modern hypertext systems (e.g Cavalier & Chandhook 1991), and Running Texts in particular, seek to re-introduce.

A contrary problem arose when the amount of gloss for certain sections exceeded the space available. In some cases this could be solved by giving the gloss an 'L' shape, allowing it to go over, or below, the main Bible text on that particular page. This was popular in the first half of the twelfth century, but from the mid-century on they preferred to continue the incomplete gloss onto the next page. Another solution, for sections that were heavily glossed, and one that became common from about the 1130's, was to divide the gloss columns into many narrower columns in which different glosses could be written. Again this is a layout that has been used in a hypertext system by Cavalier and Chandhook (1991), which has different columns to handle each of the various tasks the reader and writer might undertake.

Another solution to unbalanced levels of glossing came into favour from the beginning of the second third of the twelfth century. This involved changing the relative size of the central text column. Previously this had been of a fixed size throughout the manuscript. Now it was changed, so that in a part of the manuscript which was heavily glossed, the gloss columns became wider and the text column narrower. For areas of little glossing, the reverse was true. This procedure is similar to the common technique in graphical user interfaces, in which the size of a window can change to fit the text, or vice-versa.

As we have seen, the layout of the Biblical gloss has been specifically employed by Cavalier and Chandhook (1991) in the design of a collaborative writing system. In their system, Prep Editor, the comment (gloss) columns can be used for a variety of purposes: draft planning of the finished text, comments from co-authors or editors. At a latter stage, they envisage that it could be used in an electronic journal to provide the reader with a space to make notes and possibly engage the author in online dialogue.

> As with the Oxford gloss, there are essentially two axes along which information is connected. On the one hand, vertical columns allow users to read top to bottom connecting separate chunks of information within one category. On the other other hand, cross-referencing between separate categories is facilitated by horizontal linking with clear visual associations between related informational chunks in the plan, the content, and the comment columns (1991: 194).

This gloss interface option is appropriate when the areas of free text are small, a few sentences for example. But one can imagine an instance of the document where a small part of the computationally generated output (just a line or two) is related to a part of the conventional text which is substantial in size, a few pages possibly. In such a situation it might be advantageous to employ the mechanism used in Mathematica. That is, the free text can be compressed into a 'cell'. This is given a title and can be opened by the reader, but may also be closed. In such a scenario the gloss can be seen as performing as a sort of Table of Contents. This design option also bears a closer relationship to the structured hypertexts, such as SEPIA and gIBIS, in that elements of texts are separated out. But unlike these systems, they are linearized and may include other free text paragraphs between the cells. The notion of Mathematica cells, which can be opened or closed to reveal deeper levels, has also been used in the hypertext system Guide, to avoid the problem of having complex maps of linkages that the user must navigate (Brown 1989).

An alternative interface design for Running Texts would be similar to the Mathematica approach, in which the formal and textual components are placed one after another in a strictly linear fashion. This design option as applied to knowledge-based was unwittingly suggested by Clancey (1985b). In a section which explained the relationship between his formal NEOMYCIN model of diagnostic reasoning, based on a written protocol, Clancey interlaces the NEOMYCIN representation, which precede the relevant paragraphs of conventional text from the protocol, and which they explain. Clancey, however, did so to help explain the working of the NEOMYCIN program to other researchers, rather than seeing it as a pointer to an alternative knowledge-base interface.

Currently, the design implemented is the simple Bible gloss format of one gloss next to free text. The following section describes the actual implementation.

The ContextPane

The conventional (free) text pane is known as the *ConTextPane*. This term is both an abbreviation of 'conventional text pane', and the content of this pane also fulfills the role of the word 'context' (from the OED) as "the connected structure of a writing or composition; a continuous text or composition with parts duly connected". It is the space where the answers from the various threads are connected together into a continuous text for the reader.

The author can also select a part of the text (which have to be a set of lines) with the cursor, and specify that it be connected to a part of the gloss.Those parts of the conventional text that are to be associated, by the author, with the gloss are known as *lexia* (following the use of the term by Barthes 1975). The author and reader can identify which parts of the text are lexia by using the menu option "Show Lexia". When selected, the text of the lexia are displayed as uppercase, otherwise they appear as normal.

The actual texts of both the 'conText' and 'Gloss' texts are composed of a series of blocks. Initially, each is one simple block of text, but as areas of conText and gloss are selected and linked, these blocks are broken up into smaller blocks, the new blocks being the lexia and the surrounding non-lexia text. The same principle applies to the gloss. The conText differs from the gloss, however, in that the text can be edited. This occurs through a seperate Block Editor which can be invoked from the menu, after placing the cursor in the block to be edited.

The GlossPane

The GlossPane contains the text that is taken from the answers produced by the various threads. The text of the answer can be pulled into the TextEditor, whereupon it becomes a *gloss*. Those parts of the gloss that are selected by the author to become connected to lexia in the ConText are called *tags*. Based on these links between tags and lexia the system automatically edits the spacing of the gloss, or even the order of some of the elements, to allow the typical gloss form of alignment with the conventional text in the ConTextPane. For the reader, simple spacing is unlikely to cause any difficulty in following the line of reasoning, unless the spacing is very wide, and difficult to scan (it is in such a situation that the Mathematica interface has benefits).

Creating a Text

To create a Running Text the author has, essentially, to string together a series of Thread answers. For explanatory purposes the example below only uses the one thread created above, but more usually any significant text would contain a number of different threads. The example is based on the section of Chappell & Willis's *The Architect in Practice* (1992) dealing with contract selection (pp. 168-171). It includes a flowchart to guide the architect in deciding which contract form to select, based on a number of criteria. A (slightly simplified) form of this flowchart has been implemented as a rule-base system.

We imagine the case of an architect who has to write a document justifying his choice of a contract, a document that is used by the rest of the office, and possibly by sub-contractors. In the example, he or she would call up a new

TextEditor, an empty two pane window. In the right hand pane introductory free text might be typed, as in the title and first paragraph in the next screen image. The author would already have created an appropriate thread to explain the rational for the decision to go for the JCT 87 contract. This thread is then loaded into the gloss pane (left-hand side) by a command 'Load Thread,' which lists the available Threads and, on user selection of the desired thread, places its answer into the gloss pane. Using the answer as a template, the author then writes conventional text in the ConTextPane, on the right.

To aid the reader in seeing what was actually intended by the author, the links are now made explicit for those associations that the author did want. This is achieved in the following manner:
- Using the mouse, the author selects the line of the gloss (tag) and the lines of the ConText (lexia) that are to be linked.
- When both are selected, choosing the command 'Link Tag & Lexia", option from the menu will create a dictionary entry linking the two.

The author does this for all the links required. This takes some effort, and requires some justification. If it were the case (as occurs in this example) that the glosses and their associated glosses are strictly linear, then it would be simple enough to have the Gloss and Context permanently aligned in a simple window pane. But it is conceivable that one thread may cross over, or sit within, another. And as the current system design gives the reader only one thread gloss at any one time in the GlossPane (due to screen space problems of displaying multiple glosses at the same time), we need a mechanism to allow for the case where there are multiple overlapping threads. So the links are used by the system, which depending on the area of interest in the ConText (indicated by the user through cursor position) to place the appropriate gloss in the GlossPane so that they align with the relevent lexia.

In the figure 6.4 below, the author loaded the thread, which has been used as the basis for writing the memo in the conText pane, and a tag/lexia link is just about to be made. In figure 6.5 all the desired links have been made and the gloss has been redrawn, using the "Redraw Gloss" command, such that the tags in the gloss align with their associated lexia.

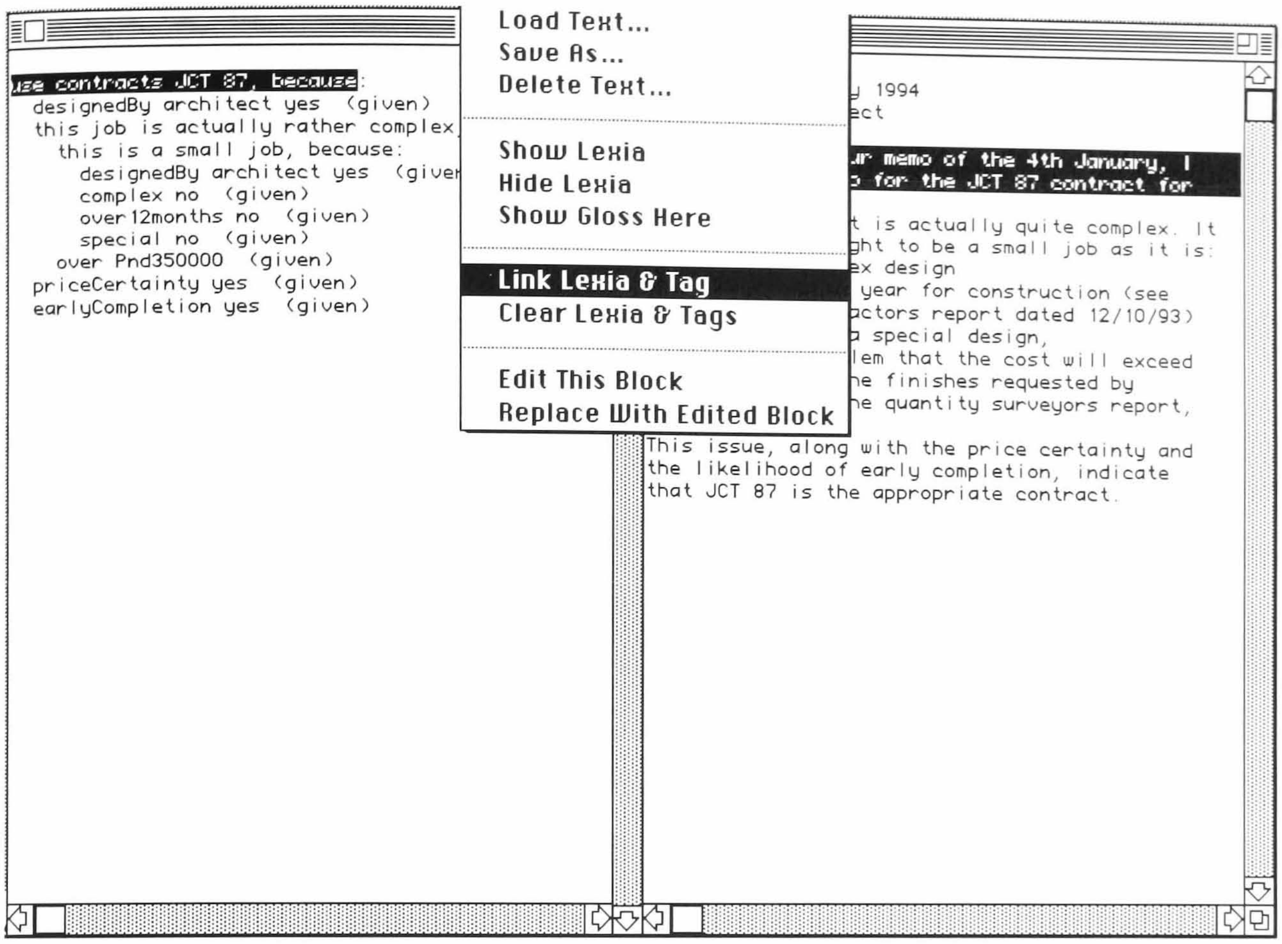


Fig. 6.4 Running Text for Contract Selection Memo

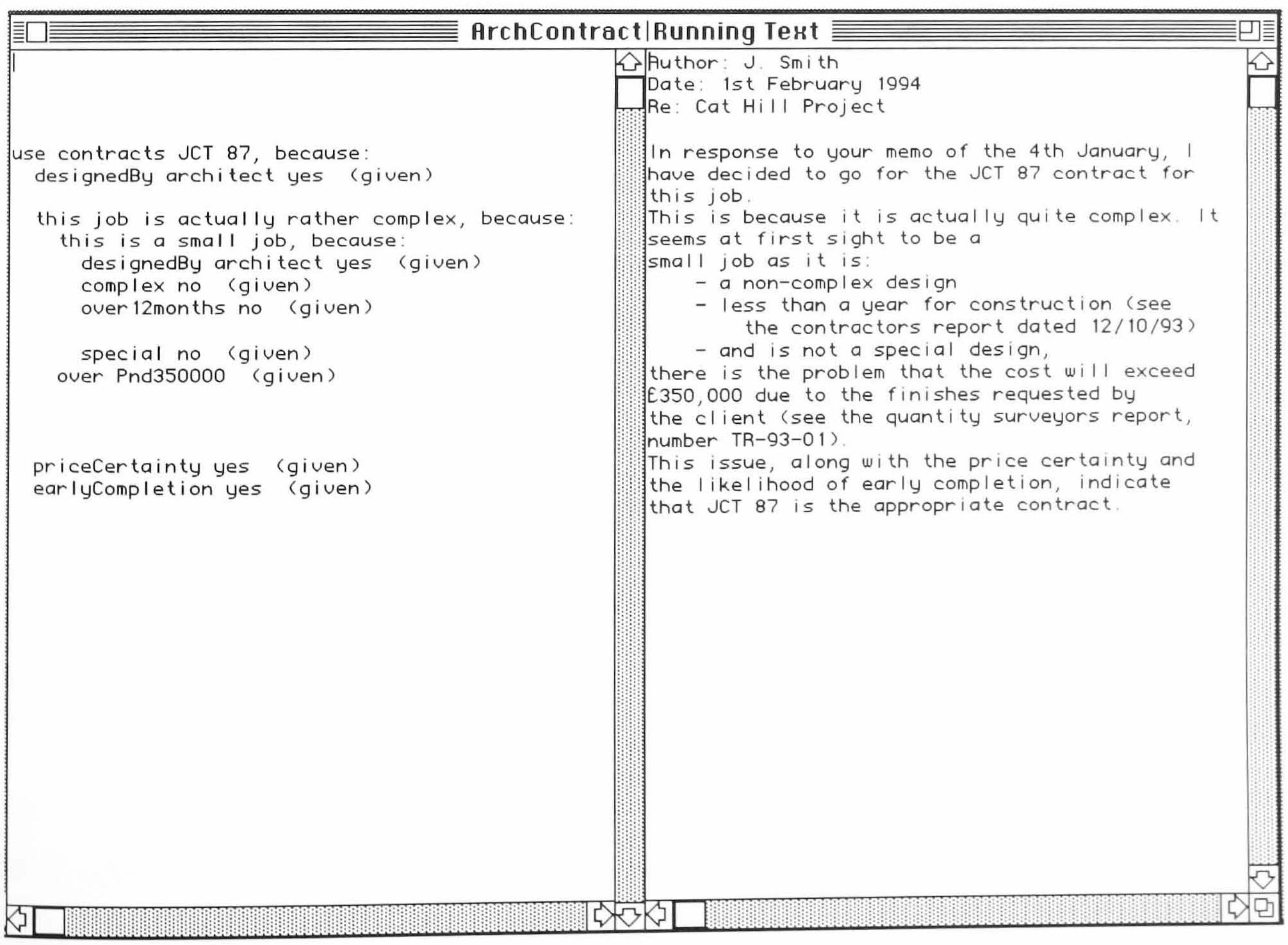


Fig. 6.5 Final Running Text for Contract Memo

The author can continue in this manner, loading in additional threads, writing the conventional text and making the links between them.

Reading the Text

Once complete, the text can now be made available to readers. If required the conventional text component could just be printed out on paper and distributed. One aim of the Running Texts is to have an evolutionary approach. It could just be used as an aid to writing texts initially, a complex text processor. Further benefits accrue, however, when the text is distributed in its computational form, be it by magnetic media or by e-mail. The then dynamic properties of the text can be exploited by the reader, to aid learning and problem-solving. Specifically the reader can:

- Increase his or her understanding of the reasoning involved by exploring the threads.
- Use the threads to assist in critiquing the text, challenging the underlying rationale.
- Appropriate the threads and text to create new texts, or *palimpsests* (this is the term for manuscripts from which the original writing has been removed in order to create space for new writing, common in the days of expensive parchment, which one did not just throw away. Often parts of the underlying text show through).
- Appropriate the threads to gloss another existing, conventional, text.

Our hypothetical reader would initially call up the Running Text. A Text Editor appears with the conventional text in the ConText Pane, while the Gloss Pane remains empty. The ConText Pane can be read in the normal way, if it that is all the reader requires. A reader can, however, go further and request to see those areas of the ConText linked to any thread, through the 'Show All Lexia' menu option. This converts all lexia within the text to uppercase, enabling the reader to see parts of the argument that have a formal correlate.

A reader might want to follow one particular point in more detail, seeing the thread from which a sentence in the ConText is derived. To achieve this one of the lexia selected can then be chosen, by placing the cursor within it and then selecting the 'Show Thread Here' menu option. The associated gloss then appears in the left-hand, gloss, pane. Using the "Redraw Gloss" menu option, the tags and lexia can be aligned. Reading the gloss, and the text that lies beside it, gives the reader a deeper view of the argument structure. If the reader wishes to know more, there is a menu option 'Invoke Editor for this Thread', whereupon a Thread Editor containing the appropriate thread appears. This allows the reader to see the specific rules in the model and all the facts declared in the query. Should the gloss be unclear, due to wide spacing, it also presents the reader with the unmodified answer.

At this point we can see how the Running Text relate to the structured hypertexts discussed in the last chapter. Like these, the conventional text is broken into a set of 'blocks', some of which are associated with the tags in the gloss. But the links between the tags (and hence the lexia) are not arbitrary, but the product of their common origin in the query and model. Unlike systems such as SEPIA, however, the Running Text is more flexible, as there is not a limited set of link types from which the author must select, but a variety of links types generated by the structuring mechanism of the AI method. The introduction of new thread types allows new structuring methods to be made available to authors and readers.

Appropriating the Text

The reader may be happy with simply reading the text in the ways described above. This is the simplest case if, for example, knowledge of AI techniques is not too deep. But the most interesting cases of Running Text readership are those of appropriation, which occurs when the reader takes the threads that are within the text and bends them to his or her own needs. Being an implementation, the strength of the Running Text system is that one can edit the query and/or model, and ask for the gloss to be recreated. This can be done for two reasons:

- to interpret another, existing, conventional text.
- to create a totally new text, based on an existing gloss.

Interpreting an existing Text

In this instance one takes an existing printed text. An analysis of the reasoning embodied in much of it suggests a thread type which might be used to model parts of it. This is the case for one example to be discussed in the next chapter. The inference engine obtained from Digitalk (1991) had a sample rule-base for tree classification, in which features can be used to place a tree into its classification. The use of a rule system for classification (a common use in the literature) suggested that it could be used to create a Running Text for a handbook on porcelain ware identification (Miller & Miller 1991).

Creating a palimpsest

Another, possibly more interesting case occurs when the reader can become a re-author, creating a totally new text by appropriating threads from an existing text.

For example, in the architectural contract text above, a reader of the document may be aware of information that the original author may not have known, namely that the price of the building will be less than expected. The reader can, having the original Thread Editor on screen, now use the menu option 'Copy Thread To...', which creates a copy of this thread in its own Thread Editor, and with a new name requested from the reader (We now have a problem in terminology, as our reader is slowly becoming an author). In this case the reader/author is aware that the cost will be below the £350,000 limit, and so changes the relevant fact in the query. The new author can now edit the original thread: reseting the facts and query, and inputting the facts again, but replacing the fact that the cost of the project exceeds £350,000 with one that puts the cost just below this level. Knowing this, the author can either use the menu to go through the alternative contract options until one can be proven, or equally likely they would already know (the inference engine used in the example does not support forward chaining).

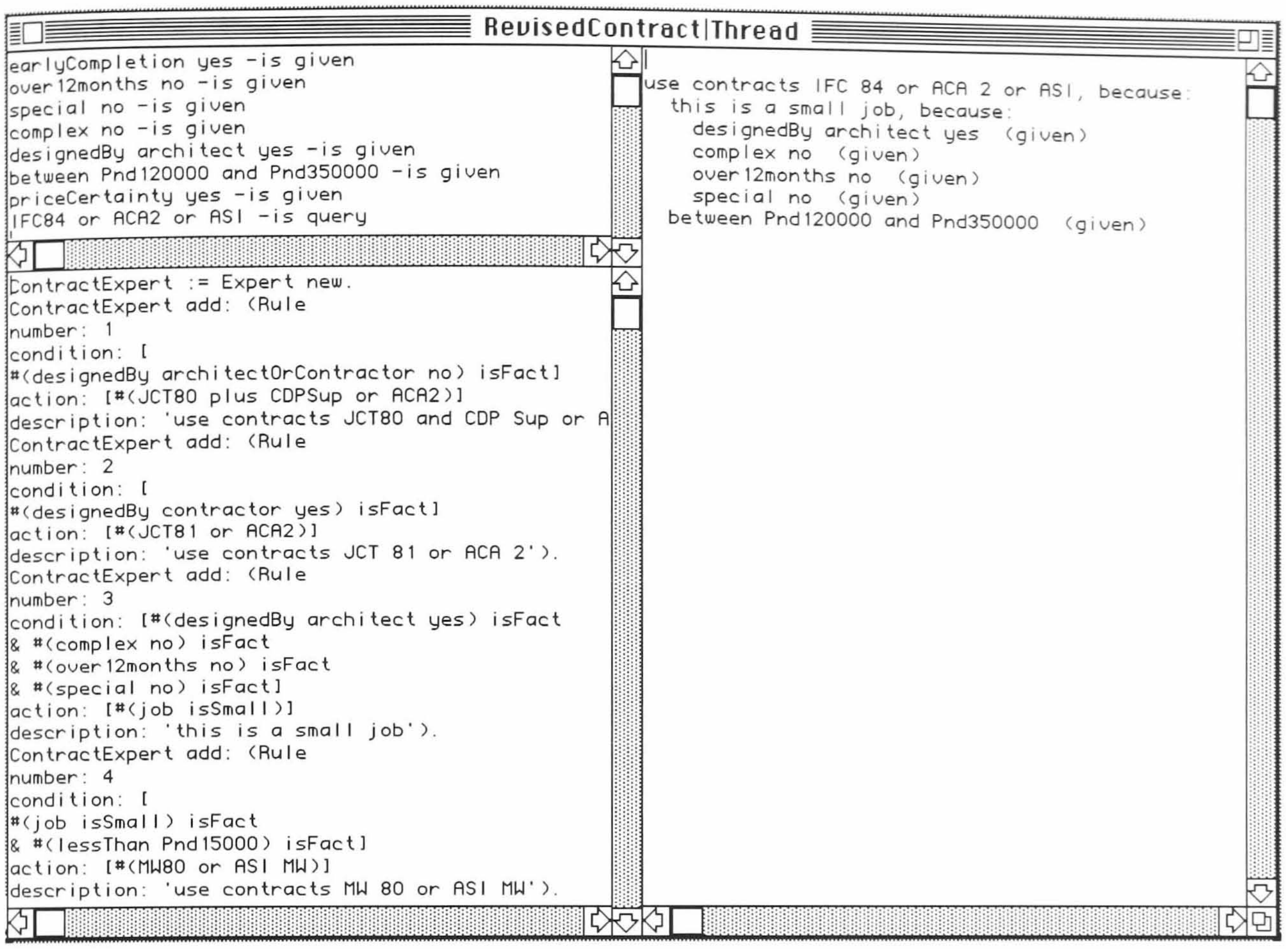


Fig. 6.6 Revised Thread for Contract Selection

Having created a revised thread, the reader-author can now edit the original Running Text, as in figure 6.7 below. The second author has taken the original memo, whose text has been put into quoted form, and added a comment to the effect that a change of policy has been made due to new information. The new thread has been loaded into the Text Editor, and the links to the additional ConText ihas been made. Any further readers can now see the reasoning, and continue to add to the argument. Of course, they could make more substantial changes. The change made above is to the query, and involves modification of one of the facts of the situation and the goal to be proved. The system also allows deeper modification. It might be the case that the very model is now inadequate, possibly due to substantial inflation in the economy which requires the limit levels to all be adjusted upwards by 10%, for example. This would involve editing the rules in the model. To do so goes deeper into the mechanism, and the reader would have to go into the rule-base. In the case of a 10% rise, the reader would have to scroll through the rules visible in the model pane, and add 10% to the values of those rules that relate to the project costs. He or she can then use menu options to either "Update Model", which will modify the rules of the existing model; or "Save Model To...", which will allow the reader to create a new, named, model that includes the changes.

The ability for Running Texts to be used in this fashion extends the value that they offer as texts. As well as being texts about a subject, they provide the reader with a set of resources that they can appropriate and exploit for other uses, maybe quite different from the original text, just as the example above was created by copying and editing the rules for a tree classifier.

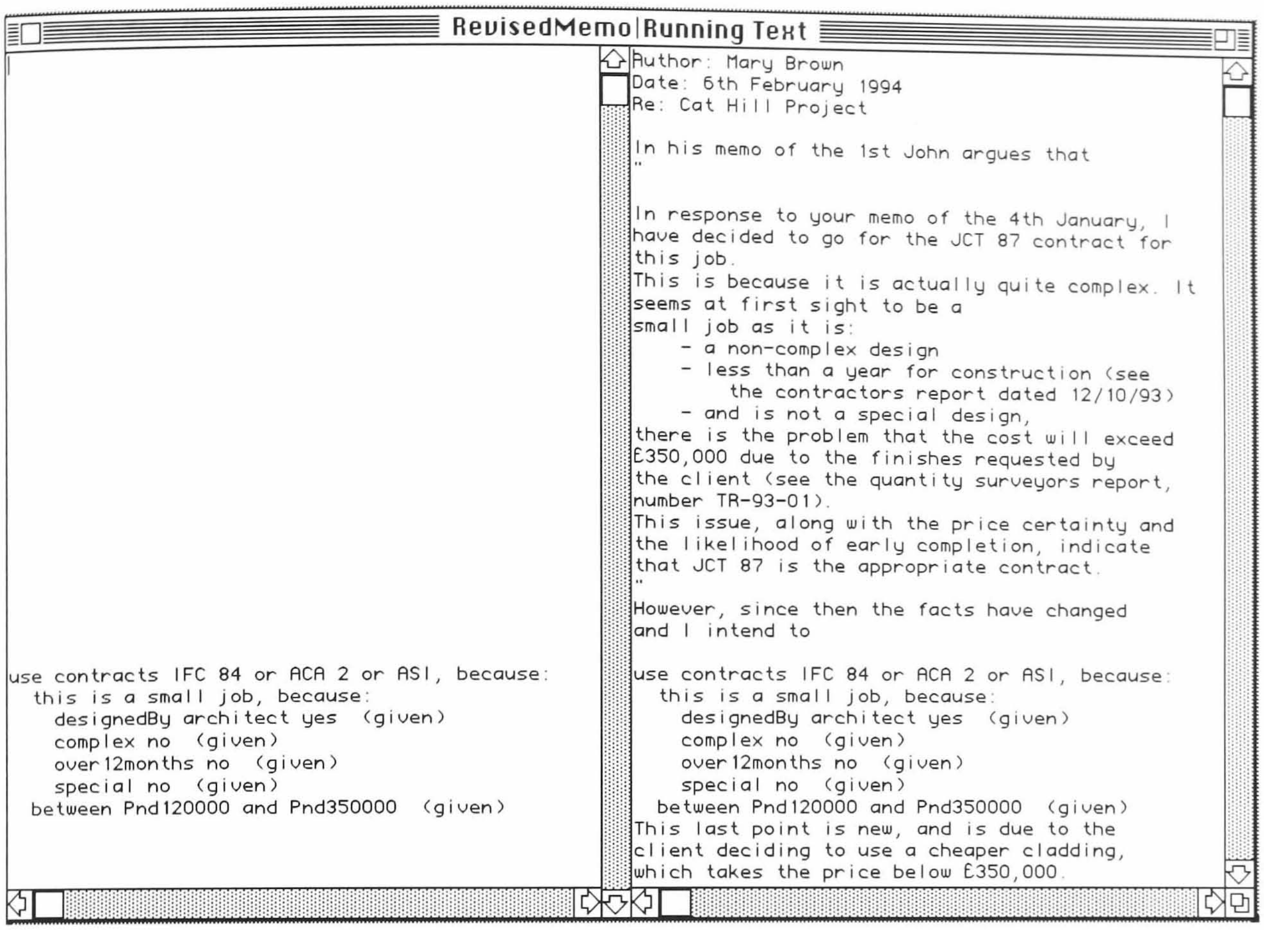


Fig. 6.7 Revised Running Text for Contract Memo

The value of being able to reuse material to address new issues, and create new texts, has been observed by Fischer et al. (1991), who developed the structured hypertext system based on IBIS described in chapter 5. They plan to direct future work into 'reusable domain-oriented issue bases', in which the issues and answers documented for one design project can be used as 'seeds' for new design projects, rather than starting with a blank sheet.

Withered Text

One area of particular interest is the notion of *withered text*. In my own use of the system I found that I had a tendency to copy over the gloss onto the ConText pane, as this was often easier and faster than rewriting the same material into conventional text, especially when I was going to be the sole reader. The conventional text came to be similar or identical to the gloss, it withered away. It is not difficult to imagine that readers would slowly come to recognise the style that certain glosses have, and that this approach might become the norm. One sees novel typographic and textual developments in printed texts already, especially within the academic arena. For example, a conference paper by Huuskonen has the following:

> We have modelled explanatory situations with contexts, resembling the viewpoints of [McKeown85]. A context in our system is made of several dimensions (1992: 350).

Notice the way that the author has used the bibliographic reference — [McKeowen85] — as part of the sentence despite its unconventional typography. Such a form is identifiable by the kind of reader that Huuskonen expects, a researcher, but in a more traditional book would probably be quite unacceptable at present.

As in many aspects of the Running Text design, withered text also has precursors in the history of writing and print. The original rationale for having the Gloss and ConText in parallel was to follow the way that writing seems to use speech as a 'crutch'. Increasingly, though, people are moving away from a strictly spoken expression in their writing. For example, initially, medieval readers used to vocalise what they were reading to assist in its understanding (Clanchy 1979: 214). Slowly this diminished as silent reading developed. The need for writing to be supported by its parent, speech, fell away. In medieval law courts, inspecting a document might mean having it spoken aloud. Literary works are frequently and explicitly addressed by the author to an audience, rather than a reader. For example we read in one work:

> If anyone wants to know this history
> Let him read the *Brut*, he will hear it there (Clanchy 1979 : 216).

Today we would write that 'he will see it there'. Reading used not only the eyes, but also the tongue, mouth and throat. Thus the system of punctuating and abbreviating Latin was aimed mainly to assist reading aloud, later being used to represent structure.

> Consequently, reading was conceptualized primarily as an oral phenomenon in the Middle Ages; the medieval mind cycled even the written word back into the oral realm...This oral emphasis made silent reading practically impossible in the Middle Ages. Silent reading was so unusual that it provoked comment (Troll 1990: 107-108).

Troll agrees with Chaytor's view, that the only explanation for private niches or stalls for reading and writing is that the noise from reading aloud while copying distracted other monks. Now, of course, adult readers can scan a document very quickly without having to have the text fully mediated through vocal expression.

There is an additional analog: between the use of the formal structures in Running Texts and the history of mathematical notation. As Cajori (1928/74: 426) records, the first printed edition of Euclid's *Elements* and early translations of Arabic algebras into Latin contained little or no mathematical symbols, employing conventional writing. Later, during the sixteenth century European algebra developed symbols for the writing of equations, but also included were the arguments and explanations of the various solution steps, written in conventional verbal form. The seventeenth century saw a culmination of desire to reduce verbal writing. In England, William Oughtred translated the tenth book of the Elements into a largely symbolic language, causing some of his readers to complain about the excessive brevity of the exposition.

The specific model of the Biblical glosses used by Running Texts have also exhibited similar trends. Later versions of glossed Bibles often reduced the amount of actual Biblical text to nothing, assuming the reader knew the text, leaving gloss alone (de Hamel 1984). These were strictly not glosses, but the text of many glosses strung together. This was fine for competent readers, who knew their Bible. However, less experienced scholars needed additional assistance to read these glosses. So within a few years, some scribes began to expand the catchwords or *lemata* in the continuous gloss itself so that, effectively, the Gloss incorporated most of the biblical text. It was disjointed, of course, and interrupted by substantial passages of gloss. A distinguished early copy is Durham. III. 10, the second volume of Gilbert on the Psalms. Although the pages appear at first sight to consist of continuous gloss, the Biblical text, written in the same sized red ink, is incorporated into it (de Hamel 1984: 19).

The 'withered text' has many of these features. The source text, the ConText, starts to fade, but has significant elements remaining to allow the reader to know where he is within an argument.

Expertise and Structuring the Workspace

Originally, it was considered that the Running Text system would be used to write textbook or reference book materials. But it soon became clear that it could also be applied to the traditional area of expert systems, which operate within a working environment, aiding users to complete an actual task. The notion brought out by using Running Texts is that one role of expertise is to structure the working environment. That is, the ability to work is enabled by giving a structure to the external environment, as well as the mental world. It is partially in the environment that representation occurs, a tenet of situated cognition. For example, Clancey (in Sandberg 1991) describes how he prepares for a camping trip. He normally goes to a box, buried in the cupboard, from his last trip. Seeing some things in the box reminds him of others. Planning does not occur only in the mind, but in the world and this external world can include texts of various forms.

An instance of such an analysis for real world tasks can be found in the work of Hutchins (1990), who has investigated the how large ships are navigated by a team of people working together. This group collaboration is facilitated by the use of tools which are not agents in the conventional metaphor, but are means of representing the task domain. That is:

> the technological devices are better seen as media for representation than as amplifiers or surrogates for cognitive abilities (1990: 218).

And in the domain of knowledge workers, Stefik et al. (1987) reports the use of a chalkboard found in the Laboratory Biology in Cambridge. It would be usually found to be covered with logic trees, and a new result will have beneath it a series of alternative explanations. Beneath these explanations came a series of suggested experiments to test them. These trees grow during the day as various people come into the office and contribute to them. They act as a group external representation guiding the work of the laboratory.

Knowledge-based systems used in a working environment, as in the architectural practice above, are a means of structuring the environment to facilitate the task. Many people use their desk in this fashion, it guides and structures their working procedures. Writing and computing systems extend this into areas of the task that previously belonged to the mind alone.

A Brief Review

Having described the workings of the Running Text system, we shall briefly review how the system addresses many of the points that have arisen in the first six chapters.

Firstly, the system allows the readers (users) to gain access to the underlying computational mechanism. This allows them to view the assumptions implicit in the facts and rule base, and also enables these to be edited if a reader has reason to disagree. Furthermore, it does so in a non-conversational manner, acting as a writing aid. It is one of Hill's third-person systems.

Secondly, on the general issues of writing, the relationship between the gloss and the conventional text allows the readers to understand the formal notation through their own prior practice. Eventually, one might imagine that these formal forms would become conventionalised and could be read in their own right, via withered texts.

Thirdly, Running Texts extend the dynamic document types found in Clarisworks and Mathematica. But whereas Mathematica extends Clarisworks by enlarging the range of mathematical functions available to the author and reader, Running Texts extend the range by offering symbolic AI mechanisms.

Lastly, Running Texts follow the development of structured hypertexts. The text is composed of many elements, some of which are given structure, through the lexia-tag links, by computational mechanisms. It is a form of dynamically-structured hypertext.

7. Issues and Examples

This chapter extends the Running Text system design from the previous chapter, looking at issues that arise, and implementational examples that illustrate some of these issues. Because Running Texts are the product of the confluence of the techniques of artificial intelligence with those of writing, there are two principal areas of concern: those that relate to AI concerns, and those that are typographical. These will be looked at in turn.

Expert Systems and Running Text

The most widespread application of AI technology is the expert system, such as the Prospector system discussed in Chapter 2. A proposal was made in that chapter for modifying the expert system to make of it a writing aid, rather than an agent, but it was remarked that the user/reader was still unable to again access to, and modify, the underlying rules. Running Texts, through the Thread Editor allow the reader to do this, and can be used to create a text that has features of a conventional expert system, but exhibits the 'AI as Writing' metaphor. The example created is based on a simple guide to porcelain ware identification, based on a small part of *Miller's Antique Checklist Porcelain* (Miller & Miller 1991). This particular text was chosen as the source material because, being a classification system based on identifiable features, it appears, initially at least, very appropriate for an expert system.

An analysis of the Miller text indicated that the classification by features had three forms:
- There are defining features for styles and manufactures. For example, both Early and Royal Sèvres have a mark that consists of two interlaced letter L's.
- There are conditional features which apply to only a subset of the pieces from a manufacturer. Thus, in Royal Sèvres if the piece is a vessel, then it serves both a practical and decorative role.
- There are also features that are only probable. For example, Nyphenburg wares often had landscape decoration in the manner of J.E. Nilson. But his work was copied by many German porcelain factories and so this does not add to the identification.

The last form of reasoning could be represented by a probabilistic reasoner, but the chosen inference engine does not have this facility and so this element was not included in the formal gloss, but only in the ConText. That is, although it cannot be represented computationally with the inference engine available, the information is still given to the reader through the conventional text component. The conditional features proved difficult to model as the rules only allow simple conjunction, but could represented using a number of shorter rules. The problem of our inference engine not being able to handle probabilities illustrates one feature of Running Texts. In a conventional expert system, if a form of reasoning is not implemented it is lost to the user, but the ability for the author of a Running Text to also express reasoning in conventional text allows him or her to give the reader material that has either no current implementational form or one that the author simply does not have to hand.

The creation of the Running Text proceeds much in the same way as the architectural contracts example above. A typical thread, one that identifies a basin or dish from Royal Sèvres, is illustrated below in Fig. 7.1. In the following figure 7.2, another thread (for Early Sèvres) is being used to begin the writing of a Running Text.

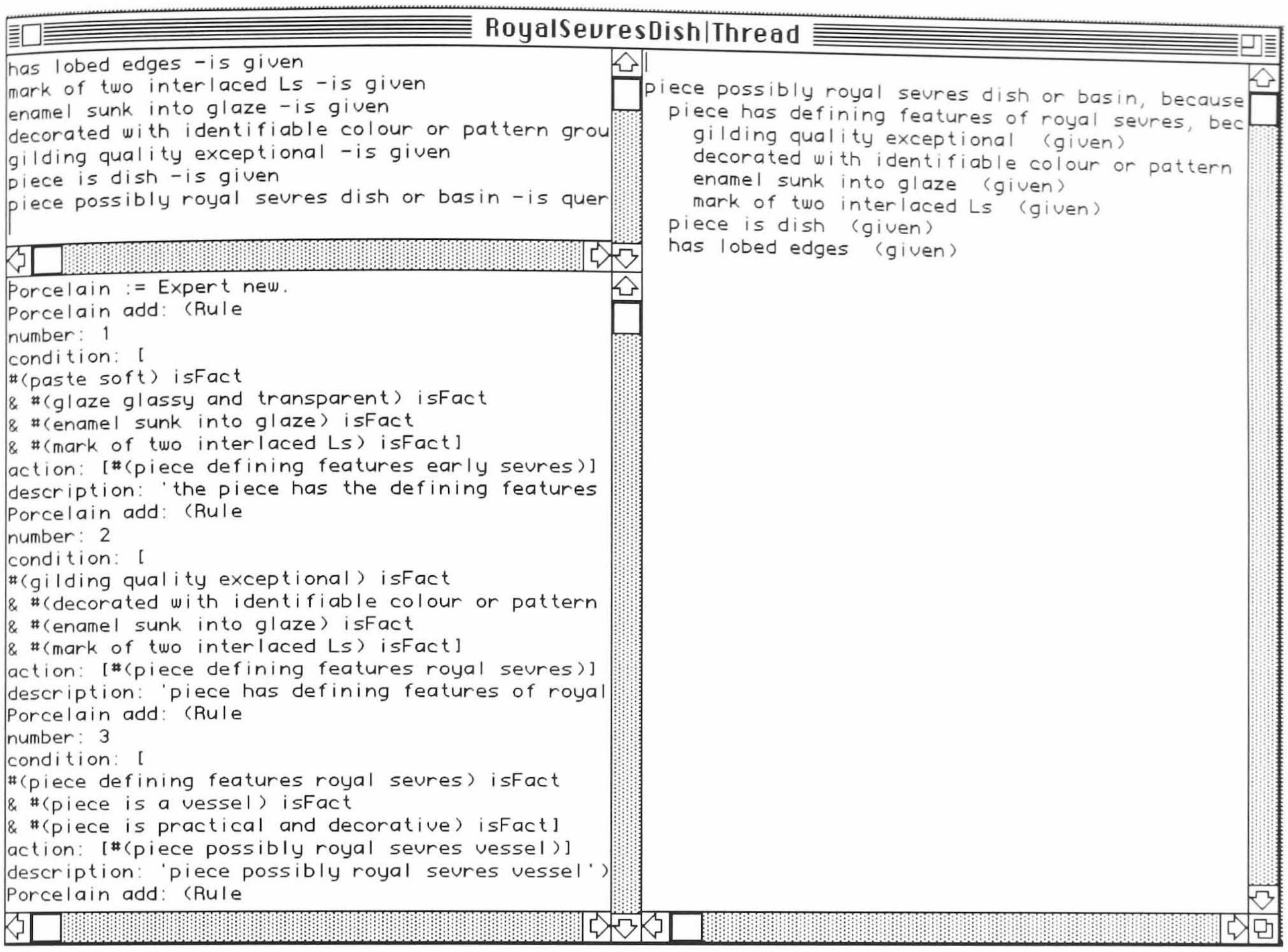


Fig. 7.1 Thread to identify a Royal Sèvres Dish

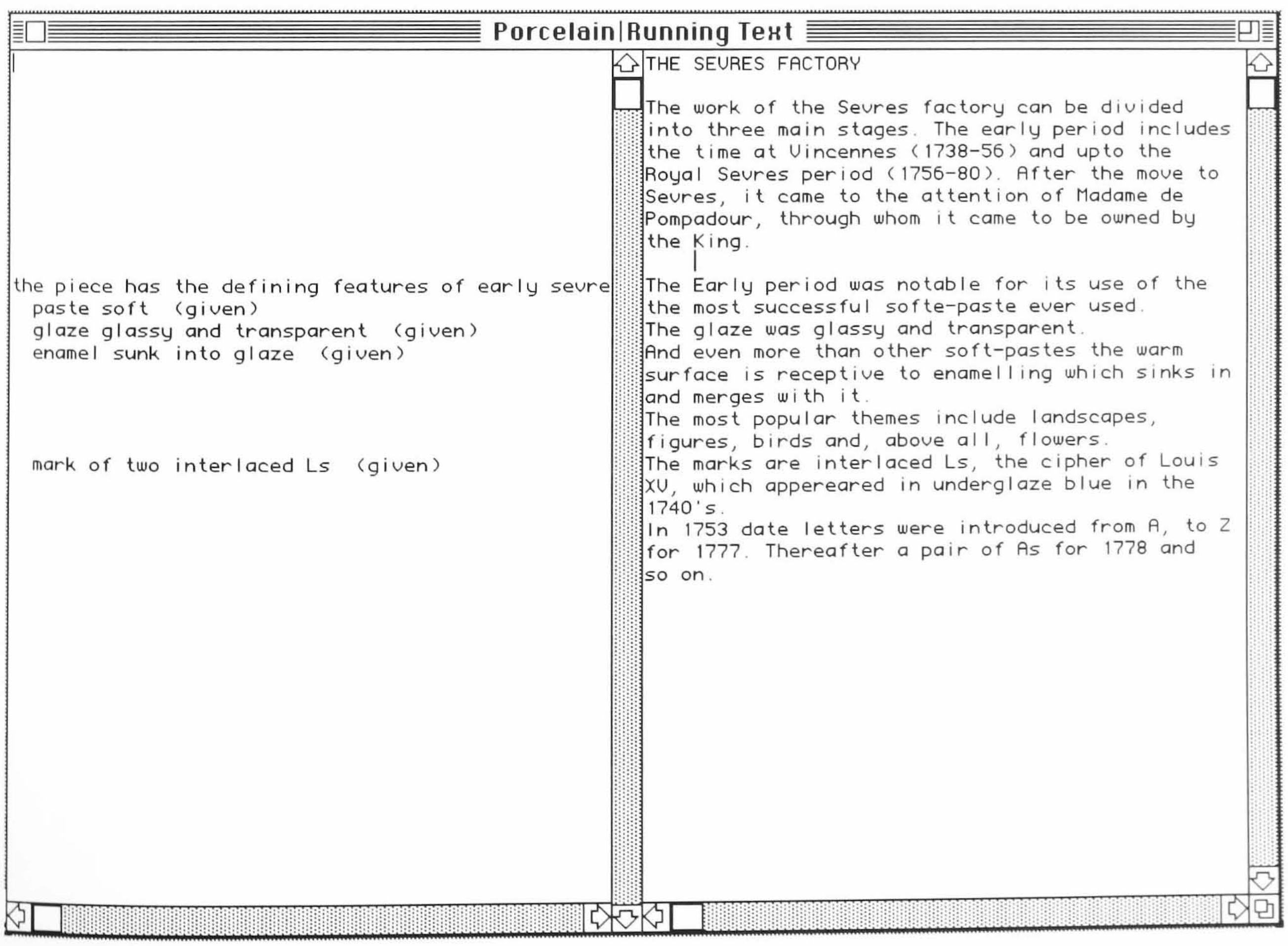


Fig. 7.2 Running Text on Early Sèvres

A set of such threads has been written which deal with the Sèvres factory which extended from its early period at Vincennes to the Post-Revolutionary period. Each period has a number of characteristics that allow pieces from it to be identified. A number of such threads can be assembled in sequence. These are assembled together within the TextEditor. This is achieved by loading in the appropriate threads, one after another, writing their associated conventional texts in the ConText pane, and linking the lexia and tags. A screen image from the final Running Text appears in Figure 7.3 below, and indicates how the Running Texts handle multiple threads.

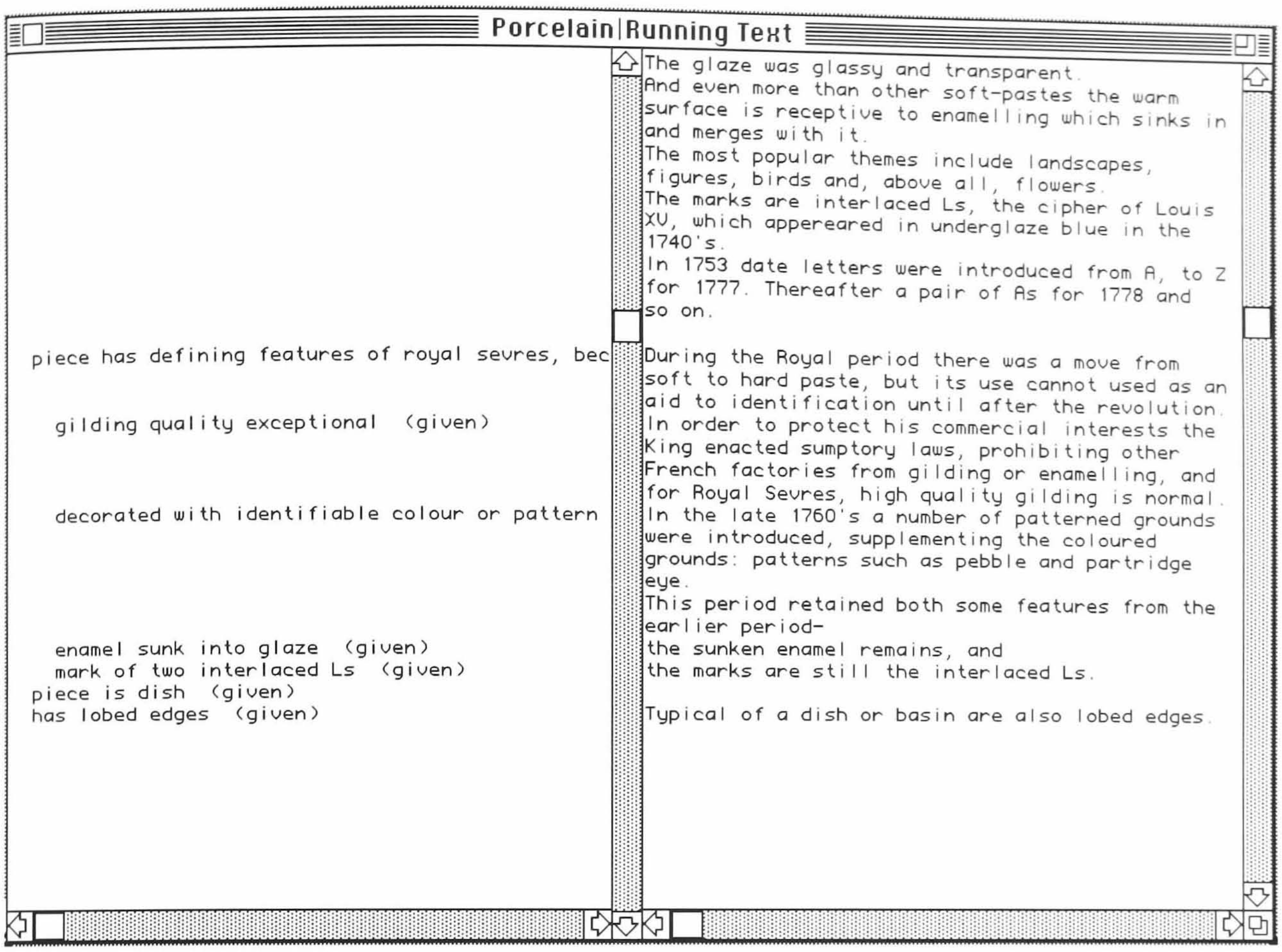


Fig. 7.3 Running Text for Sèvres Factory

This particular Running Text when complete, can, as in the previous example, be read at various levels. Firstly, for complete initiates, the ConText alone can be read as if it were ordinary text. A table of contents or conventional word-search index can help such a reader arrive at particular parts of the text, as required. At the next level, readers can now see the thread that underlies particular parts of the text and begin the see the structure of the reasoning that underlies the domain, the kinds of features that are important and how they are considered.

Secondly, as in the contracts example, readers may disagree with certain assumptions in the rules. Having access through the Thread Editor allows them to edit rules to fit their own experience or new information that comes to hand. For example, it could be that a particular feature is suddenly no longer considered as being good for identification purposes, possibly because it has been found to be used in many fakes. Alternatively, new technology may allow previously impossible features to be recognised, features that definitely identify a certain manufacturer (optical properties of the glaze for example). Appropriate modifications can be made to the basic model.

The question that arises, of course, is how to give the reader the same useful functionality as that of the traditional expert system, without those concerns that were discussed in chapter 2. In other words, how can the reader come to the Running Text with an unidentified piece and, by 'querying' the text, obtain a possible identification. It can, in fact be done quite easily. For this to be possible, the author must write additional rules into the model, which in effect, say that if the inference engine can reach a certain level of description of the piece, then this is satisfactory for identification. For example, in the part of the Running Text above, an additional rule:

```
condition: [#(piece possibly royal sevres dish/basin) isFact]
action: [#(piece identified)]
description: 'piece is identified'.
```

This requires that the author determine what level of identification is likely to be wanted by the typical reader: just the factory, or also the period, and so forth. Exactly the task that the author of any good text must consider.

One can imagine an introductory preface to this specific Running Text in which the reader is told how to use the text to undertake such a task, which would proceed as follows:

- The preface contains instruction on how to use the facility, and a basic thread in which the query is of the form:

piece identified - is query

- The reader adds facts, of which initially there are none, to the query, based on their inspection of the porcelain piece before them. This involves selecting features that match their piece from a menu list of those available.

- When the menu option 'Set Query' is selected the system will try to find a piece that matches the features.

Once a possible identification has been made, the reader can then go to the part of the Running Text that contains a ConText on the style identified, providing additional assistance in aiding the reader's identification and understanding of the piece.

Using the system in this 'expert system' fashion brings certain out new ways of looking at how the Running Texts can be understood. On the one hand, it can, as already discussed in some detail above, be seen as a structured hypertext, in which the structure is provided by the rule system. This view seems more obvious when it is used as an authoring aid: to create the contract memo in the previous chapter, for example. But it can also be seen as a self-documenting expert system, especially when used by readers in this problem-solving mode. The porcelain expert system would avoid some of the problems inherent in the traditional expert system. As the conventional text can be read first, the reader gets an idea of the scope and depth of the system. Thus, exaggerated expectations are less likely to be formed than when using a conversational expert system, and the conventional text allows the system builder to give the user access to those parts of the domain knowledge that are difficult to formalize. The Running Text format also allows the reader to add to the body of texts. That is, when a reader uses the original Running Text to identify a piece of porcelain within a professional environment, he or she can then write a new document explaining the identification and including a photograph. This document adds to the stock of knowledge available to other readers, building an archive. One can conceive of this archive as being an appendix added to the basic Running Text.

The idea of integrating hypertext and expert systems has been proposed by many (Gray 1992), but the connection is generally put forward as a means to ameliorate the problems that each may have, and maintain the position that the AI component has 'agency'. For example, Yankelovich, Meyrowitz & Van Dam propose that a multimedia system could have knowledge-bases containing profiles of the readers, and of the information it contains. Such systems would become 'assistants', "carrying on a dialog with the reader" (1985: 17). Running Texts, because of the theoretical basis that treats both AI and hypertexts as technologies of the word, as ways of writing, provides a deeper integration.

Additional Thread Types

One important task is to show the generalisability of the approach being advocated by implementing other types of cognitive method in addition to the simple forward-chaining rule method in the example above. The design of the Running Text as a medium requires that there be a sufficient vocabulary of methods to be able to 'write' a reasonably large range of document. Other such AI methods, or 'lines of thought', to be taken from the AI literature might include:
- causal reasoning (Riesbeck 1984),
- qualitative simulation (Kuipers & Kassirer 1984),
- metaphorical reasoning (Turner 1987),
- analogical reasoning (Wellsch & Jones 1986),
- induction from data (Langley et al 1987),
- the Toulmin model of argument (Clark 1991, Stutt 1989),
- graphical, spatial reasoning (Mitchell 1990).

In a design environment, the graphic threads will be of particular importance. In those tasks where design centres on the development of form, the language employed becomes so interleaved with deitic references to graphics that they are often incomprehensible without graphics. It is for this reason that recent work of applying IBIS to design has dealt with the problem of connecting this IBIS system to graphic expression of the design being undertaken, for example, in a kitchen floor-plan (McCall et al. 1990). One might imagine that in implementational terms this would suggest a graphical version of a Running Text document, where instead of text panes, the answer pane of the Thread, and the 'gloss' and 'ConText' pane of the 'TextEditor would be graphic panes. A sample query might be a to ask for those Palladian style floor plans that fulfilled certain requirements (stated in the query). This would generate a set of possible floor plans for the gloss, one of which could be copied over to the main graphic, drawing pane, and which could act as a template for a fuller drawing editing system. Links could be maintained by drawing the final image over the copied gloss (which lies on a 'layer'). Through this layering the reader could access this gloss which by links gives the reader access to the original thread.

The range of methods can be further widened, for another issue that arises is the way that by including certain, more mathematical and formal modes of reasoning, such as induction, the Running text can be seen to merge with computational hypertexts. Indeed, the join is already becoming vague. For example, Mathematica has integration functions (Wolfram 1988: 419-426), which allow the Mathematica user to undertake the kind of tasks dealt with by an early AI program SAINT (Feigenbaum & Barr 1981/82: 123-125). This program is considered an classic example of early AI work. One would imagine that such methods as these would also be made available, despite the fact that (because of their well understood nature) they are often not considered as writing within AI.

Causal Reasoning

As an example of another AI method we shall look at causal reasoning as described by Riesbeck (1984) variations of which are often used in economics (Bouwman 1983; Pau & Gianotti 1990). This method was also the basis of the work undertaken by the author as part of the DecisionLab project (Huxor 1988). This is a simple form of reasoning in which one creates a network of variables connected by directed links of influence.

This method can be applied, for example, to the kind of reasoning as expressed in de Bont, Schoormans & Wessel (1992) where the factors affecting the acceptance of designs are evaluated. They investigated how both aspects of personality (such as tolerance of ambiguity) and socio-demographic characteristics (such as age and level of education) affected consumers acceptance of new product design. A causal model was presented and tested empirically to study the significant effects and their inter-relation. They finally proposed a causal model, of which a simplified version looks thus:

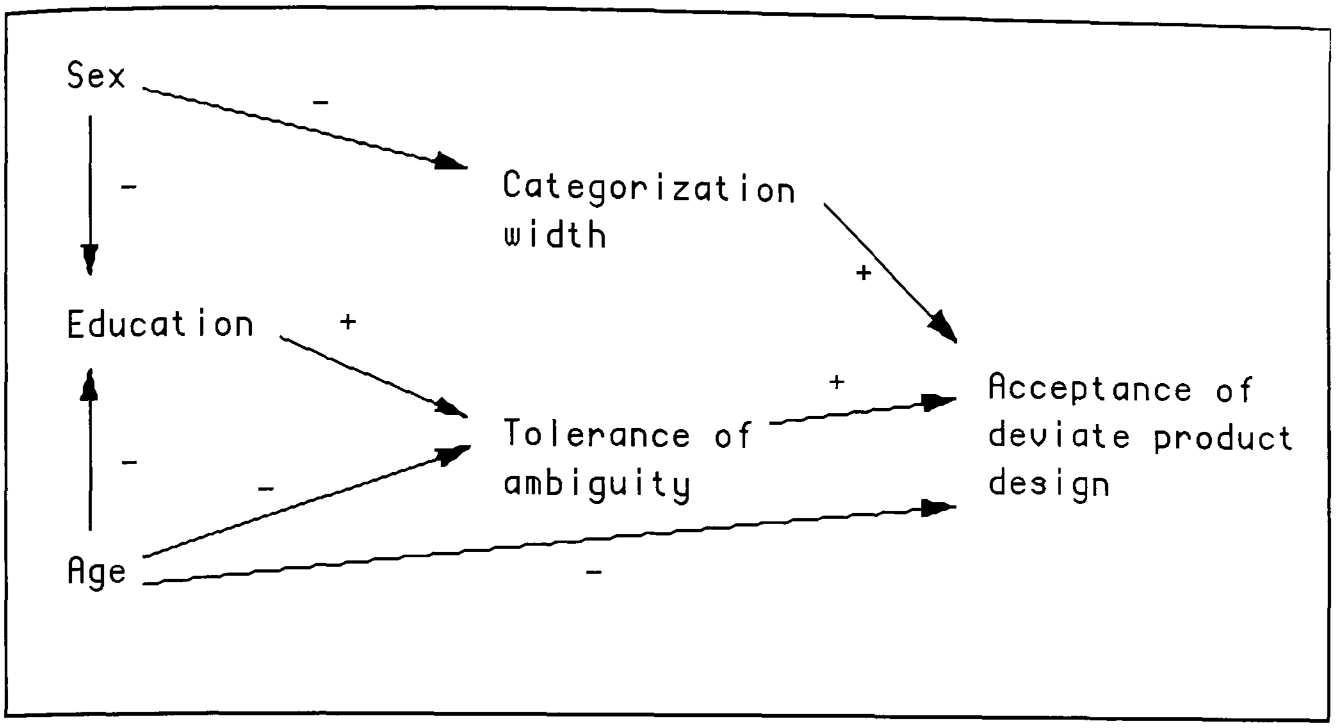


Fig. 7.4 Interaction of Variables Affecting Acceptance of Design
(adapted from results of de Bont, Schoormans & Wessel 1992)

The model has been somewhat modified from de Bont, Schoormans & Wessel's original, omitting the strength of the causal relations and retaining the positive or negative aspect only. Thus the figure above can be read as including the following: a rise in education gives a rise in tolerance of ambiguity, whereas a rise in age leads to a fall in tolerance of ambiguity (the use of sex as a variable with male being a higher value than female is a feature of de Bont, Schoormans & Wessel's original model). Some of the variables need further explanation. *Categorization width* is a personality factor that indicates how many categories a person forms to classify a given situation. A person who thinks that a Porsche is not a car, but a sports car, is a narrow categorizer. If this person were looking for an 'expensive new car', a Porsche would be excluded as an option. *Tolerance of ambiguity* is a cognitive style factor that indicates how tolerant individuals are to ambiguous situations, which are new, complex and insoluble. *Acceptance of deviate product design* is the degree to which a consumer will purchase products with a novel design.

The algorithm for simple causal path analysis has been written in Prolog/V a prolog interpreter written within Smalltalk/V. The links between the various variables in the model are represented by the predicate
link(node1, node2, linkType)
which means that node1 is connected to node2 by a like of type linkType, which can be positive or negative:

```
link(#sex,#education,#neg).
link(#age,#education,#neg).
link(#education,#toleranceOfAmbiguity,#pos).
link(#age,#toleranceOfAmbiguity,#neg).
link(#sex,#categorizationWidth,#neg).
link(#categorizationWidth,#acceptanceOfDeviateDesign,#pos).
link(#toleranceOfAmbiguity,#acceptanceOfDeviateDesign,#pos).
link(#age,#acceptanceOfDeviateDesign,#neg).
```

The causal reasoner is a simple algorithm to find all paths between a startNode and a endNode. Consider the query:

```
causalPaths(#rise,#age,_,#acceptanceOfDeviateDesign,path)
```

which can be interpreted as: "if there were a rise in age, what would be the effect on acceptance of deviate design, and give the paths." If this query is applied to the links model above, the following answer (explanation) is generated.

```
(([[rise age ] [fall education ] [fall toleranceOfAmbiguity ]
fall acceptanceOfDeviateDesign ])
([[rise age ] [fall toleranceOfAmbiguity ] fall acceptanceOfDeviateDesign ])
([[rise age ] fall acceptanceOfDeviateDesign ]))
```

The first path could be read as meaning that 'a rise in age indicates a drop in education level, which then gives a reduction in the tolerance to ambiguity. This finally suggest a lower level of acceptance to novel designs'. This answer can, as in the rule-base example, be loaded into a Text Editor and used to construct a text.

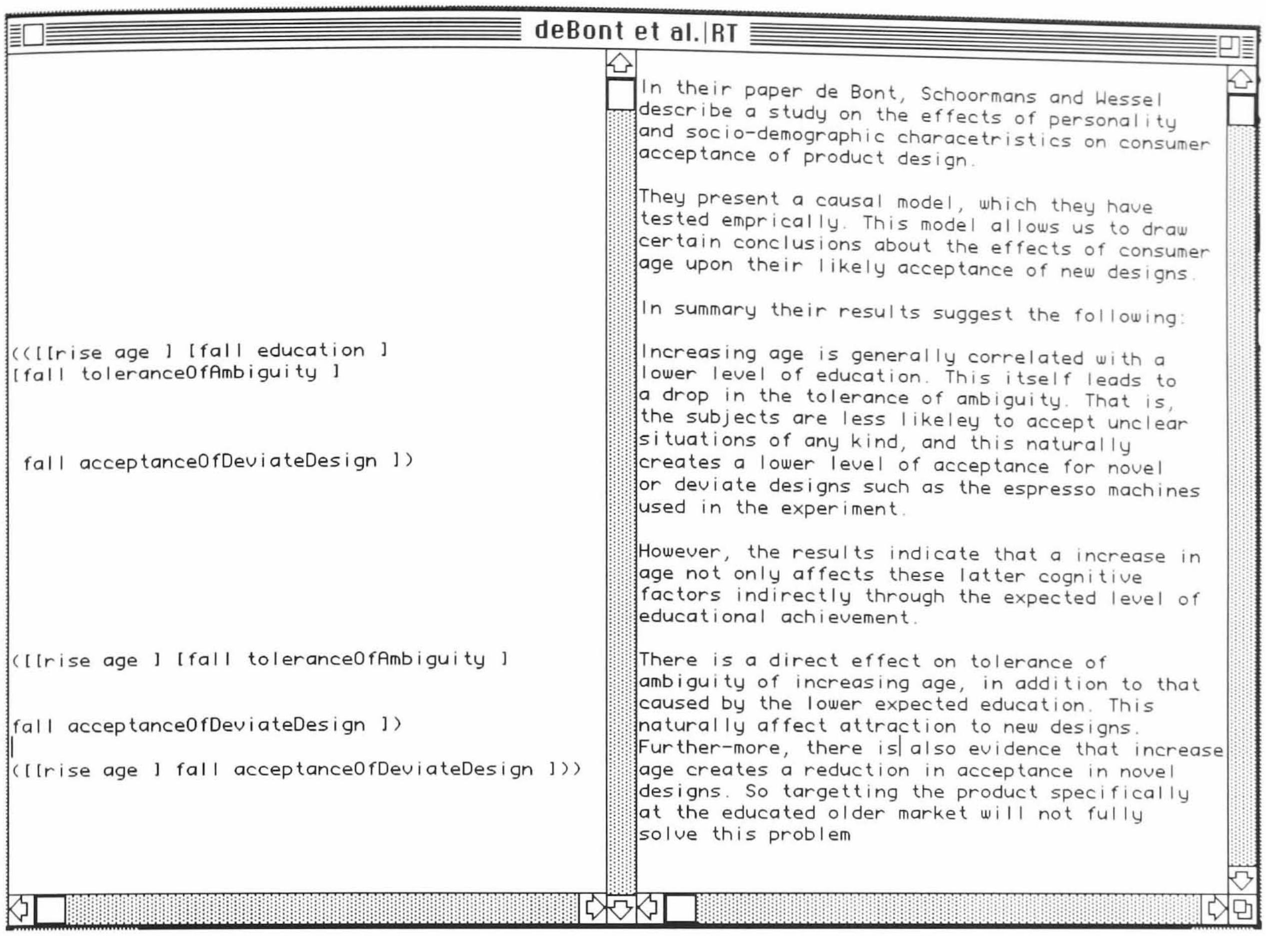


Fig. 7.5 Running Text of Causal Model for New Product Acceptance

The causal model used by de Bont and his colleagues further illustrates how AI and more conventional, numerical, computing can be brought together in the one environment. The original causal model was created by a statistical analysis package called LISREL. So a fully Running Text of the de Bont research would also include the experimental results, and LISREL would be incorporated, allowing the variables and their connections to be calculated. It begins in this situation to be difficult to see where the AI starts and the traditional data processing ends. They are both formalizations of ways of describing and solving problems.

The simplicity of the causal reasoning method also makes it very easy to appropriate for other domains. It is far simpler to edit a list of variables and connection than a set of complex rules. The linking of variables with positive and negative connections is one that is seen in many domains, and it is a simple matter to edit the basic link predicate to create a new model. Given a graphical interface, the task becomes even simpler, the author creating named nodes and linking these with arrows of either a positive or negative type.

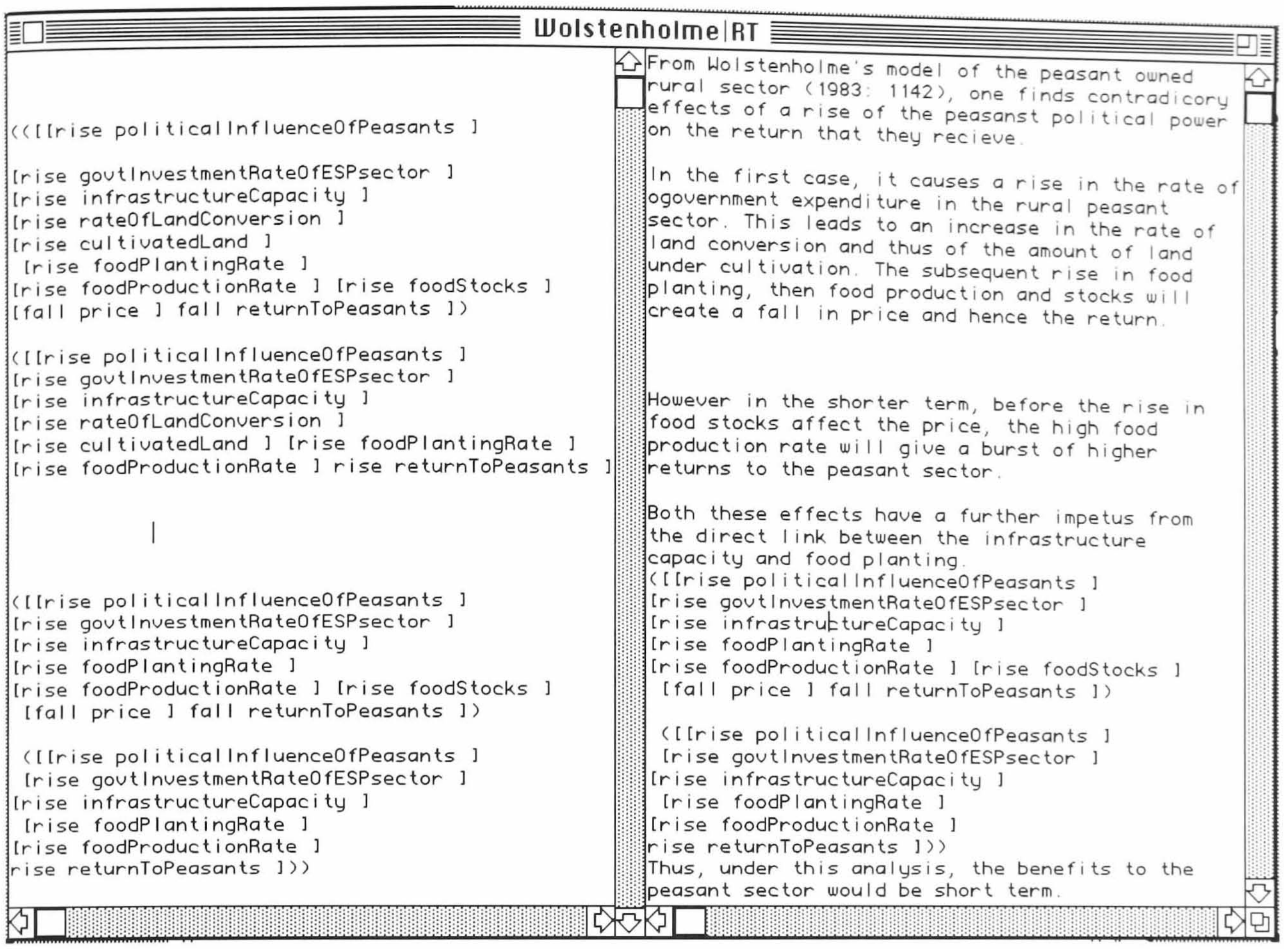
For example, causal modelling has been used to help describe and analyse national development programmes. Wolstenholme (1983) uses the notation to describe various socio-economic sectors, including the peasant-owned rural sector (1983: 1142). The relationships modelled there can be simply made into a causal model based on the same format used by the de Bont paper. The new links are:

```
link(#totalPopulation,#foodRequired,#pos)
link(#foodRequired,#price,#pos)
link(#price,#returnToPeasants,#pos)
link(#returnToPeasants,#standardOfLivingOfSEP,#pos)
link(#politicalInfluenceOfPeasants,#govtInvestmentRateOfSEPsector,#pos)
link(#govtInvestmentRateOfSEPsector,#infrastructureCapacity,#pos)
link(#infrastructureCapacity,#rateOfLandConversion,#pos)
link(#rateOfLandConversion,#cultivatedLand,#pos)
link(#cultivatedLand,#foodPlantingRate,#pos)
link(#infrastructureCapacity,#foodPlantingRate,#pos)
link(#foodPlantingRate,#foodProductionRate,#pos)
link(#foodProductionRate,#foodStocks,#pos)
link(#otherFoodProductionRates,#foodStocks,#pos)
link(#foodStocks,#price,#neg)
link(#foodProductionRate,#returnToPeasants,#pos)
link(#numSEP,#foodProductionRate,#pos)
link(#numSEP,#standardOfLivingOfSEP,#pos)
```

The query:

```
causalPaths(#rise,#politicalInfluenceOfPeasants,_,#returnToPeasants,path)
```

is then applied. This query can be interpreted as asking what are the effects of a rise of the political influence of peasants on the financial return that the same peasants receive (and show all the causal paths). The answer, when glossed into a Running Text with appropriate conventional text gives us the following screen image:



WolstenholmeRT

((([rise politicalInfluenceOfPeasants]

[rise govtInvestmentRateOfESPsector]
[rise infrastructureCapacity]
[rise rateOfLandConversion]
[rise cultivatedLand]
[rise foodPlantingRate]
[rise foodProductionRate] [rise foodStocks]
[fall price] fall returnToPeasants])

([[rise politicalInfluenceOfPeasants]
[rise govtInvestmentRateOfESPsector]
[rise infrastructureCapacity]
[rise rateOfLandConversion]
[rise cultivatedLand] [rise foodPlantingRate]
[rise foodProductionRate] rise returnToPeasants]

([[rise politicalInfluenceOfPeasants]
[rise govtInvestmentRateOfESPsector]
[rise infrastructureCapacity]
[rise foodPlantingRate]
[rise foodProductionRate] [rise foodStocks]
[fall price] fall returnToPeasants])

([[rise politicalInfluenceOfPeasants]
[rise govtInvestmentRateOfESPsector]
[rise infrastructureCapacity]
[rise foodPlantingRate]
[rise foodProductionRate]
rise returnToPeasants]))

From Wolstenholme's model of the peasant owned rural sector (1983: 1142), one finds contradicory effects of a rise of the peasanst political power on the return that they recieve.

In the first case, it causes a rise in the rate of ogovernment expenditure in the rural peasant sector. This leads to an increase in the rate of land conversion and thus of the amount of land under cultivation. The subsequent rise in food planting, then food production and stocks will create a fall in price and hence the return.

However in the shorter term, before the rise in food stocks affect the price, the high food production rate will give a burst of higher returns to the peasant sector.

Both these effects have a further impetus from the direct link between the infrastructure capacity and food planting.
([[rise politicalInfluenceOfPeasants]
[rise govtInvestmentRateOfESPsector]
[rise infrastructureCapacity]
[rise foodPlantingRate]
[rise foodProductionRate] [rise foodStocks]
[fall price] fall returnToPeasants])

([[rise politicalInfluenceOfPeasants]
[rise govtInvestmentRateOfESPsector]
[rise infrastructureCapacity]
[rise foodPlantingRate]
[rise foodProductionRate]
rise returnToPeasants]))
Thus, under this analysis, the benefits to the peasant sector would be short term.

Fig. 7.6 Wolstenholme Causal Model as a Partial Running Text

Use of 'Running Texts' within AI

The work above proposes a full use of the writing metaphor to create a knowledge medium. However, many of the ideas could be adopted by traditional AI researchers, without necessarily having to accept the full position. There are two applications that spring to mind.

Firstly, knowledge engineers could publish their results by creating deliverable Running Text type systems linking the model to original (and probably edited) text of the protocol, in addition to convention journal publication. Conventionally, in knowledge elicitation, texts are created, be they protocols from interviews with experts undertaking a problem solving task, or excerpts from the domain literature. These are then analysed and parts converted into into executable systems that operate on the dialogue model. In this new approach the work would continue in much the same way, but the deliverable would be a Running Text in which the the executable component is delivered with the full text. These documents could be sent to other AI workers, who would be able to evaluate to potential of the work more actively. The readers could then apply the representations embodied in the document in a more traditional fashion if desired.

Linster & Musen (1992) have employed the KADS knowledge engineering methodology to the modelling of the ONCOCIN expert system. They note a number of lessons, which support the use of a Running Text type system by knowledge engineers. From their experiences they suggest that the KADS researchers should address the following issues:
• There needs to be an extensive library of existing interpretation models to which the knowledge engineer can refer to build a similar system.
• Novice knowledge engineers need more extensive exemplar case studies.
It is exactly these features that a library of Running Texts provides: both a catalogue of threads for various problem types, plus context material as specific cases studies of their use.

The second application of the 'AI as Writing' is in the final delivery of expert systems, when one could modify the user-interface. As was suggested in chapter 2, the dialogue metaphor could be modified to create a simple document authoring system. The simple tool-based interface could be appropriate for applications where there is little dispute in the domain, and authors would not want to get into the AI program component.

Networks of Knowledge and of Text

In the examples implemented above, the Running Texts have been composed of only one or two threads, and in the latter case, they simply follow one after the other. However, the interface choice of Biblical style gloss was made to allow non-sequential threads. How this might function is best described schematically. Currently, the implementation allows for only one gloss to be visible at any one time. But it is possible that a particular line of argument might involve three threads, and hence three glosses, all contributing to the point being made. Cavalier and Chandhook (199) have implemented exactly such a multiple column interface, but in their case, the glosses have less content, consisting of annotations and notes. Even the one gloss and context used in the causal reasoner above looks rather crowded.

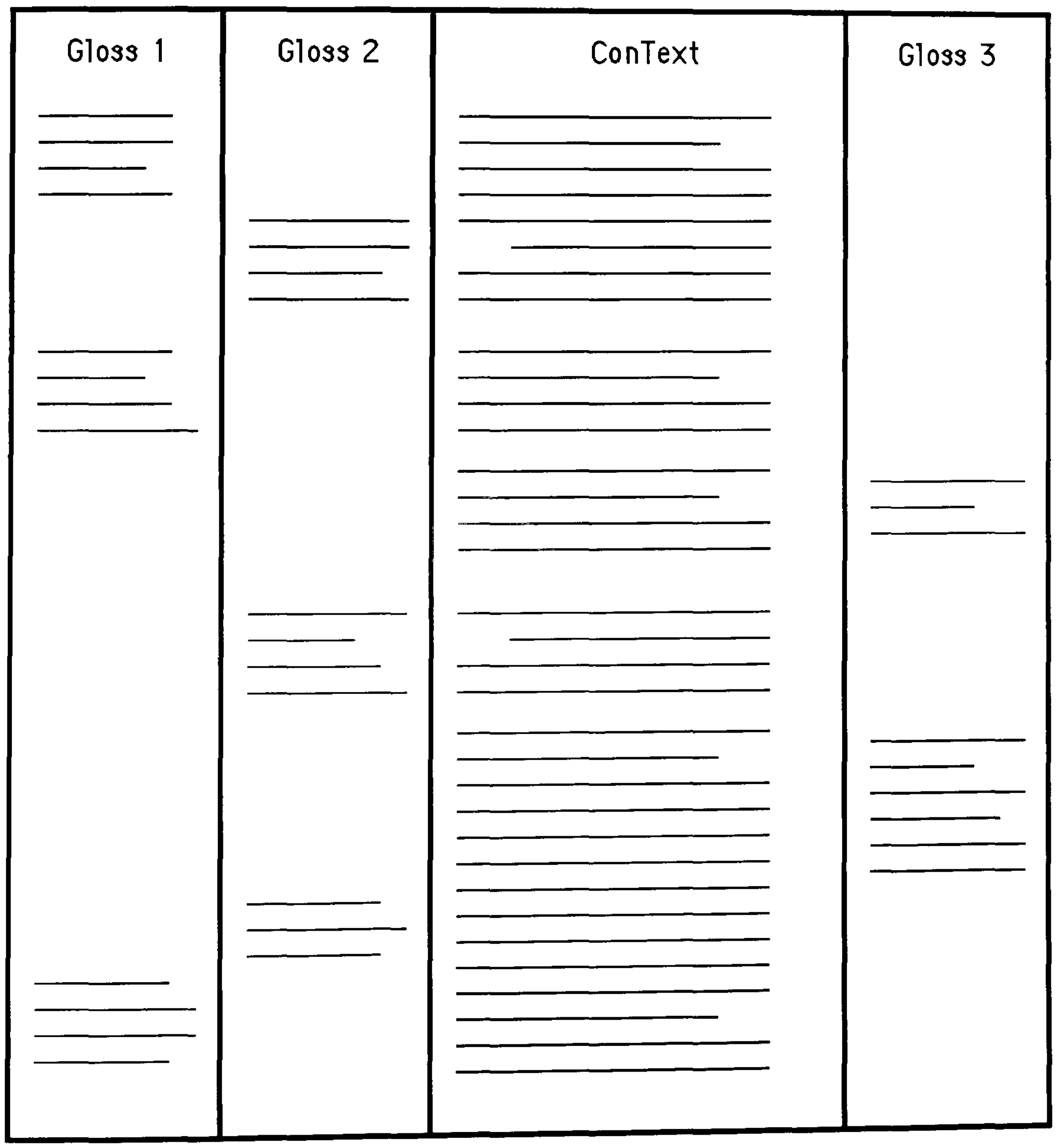


Fig. 7.7 Multiple Glosses Linking in a Context

This is an issue that must eventually be addressed, however, as the ability to have multiple threads link together in support of an argument is likely to be important. Again one finds very close parallels between the theory of text and that of knowledge-based systems in recent work on how they can be structured.

The parallels discussed in earlier chapters centred on the single word or short paragraphs. But there are also features from extended texts that can be employed to inform the design of the knowledge medium and the most important of these is the claim that extended texts can be considered as a network of threads. For example, Barthes, in his analysis of *Sarrasine*, a short story by Balzac, believes that we find in the text a "kind of network, a topos" (1975: 20). He extracts the various lines of reasoning that lie in the text, and in doing so he employs a technique that is remarkably close to the methods of knowledge engineering, breaking the text into distinct components, which he terms 'lexias' [sic] (hence the use of the term in Running Texts). These are then woven into an analysis, or to quote Miller:

> The critic adds his weaving to the Penelope's web of the text, or unravels it so that its structuring threads may be laid bare, or reweaves it, or traces out one thread in the text to reveal the design it inscribes (cited in Norris 1982: 96).

The roots of the term 'text' indicate its close links to such a view (Stock 1984/5). The verb *texo,* from which the roots for the English, German and Romance language terms for text are derived, meant to weave, plait, or interlace. A glossary of Late Latin refers to *texta* as connections or chains, and in the early medieval period words derived from *texo* dealt more frequently with cloth than with writing.

The same argument also applies by extension to any technology of writing, which will become a weaving together of verbal elements (Bolter 1989). The relationship between writing and weaving finds an expression in the Tamang culture of Nepal, which have charms that consist of a piece of paper on which the appropriate text has been stamped. This paper is then folded into a small square and carefully bound in multi-coloured threads. It is not just the binding of textual oaths that makes them effective, but also the binding of threads; nor is it just the weaving of cloth that unites the world, but also the weaving of words (March n.d.). This net metaphor is also found in West Africa. Native singers carried in a net (a form of loose weaving) all kinds of objects, such as pipes, feathers, skins, birdheads, bones, etc. Each of these served the purpose of aiding the singer to recall a certain song (Gelb 1965: 4), just as the simple pre-writings described in chapter 3 aid recall.

Examples of what Miller calls the extraction of one thread to reveal its design are the work of Turner on kinship metaphors (1987), and my own work in causal reasoning in economic texts (Huxor 1988). But what seems to be important for the understanding of any real-world text is not just the extraction of one thread, but of many, and an appreciation of their interconnectivity. Perelman makes a similar point, arguing that non-formal argument is not a chain of ideas, but rather a web formed from all the arguments and all the reasons that combine and achieve the desired result:

> the arguments we use seem to me very much more like a piece of cloth, the total strength of which will always be vastly superior to that of any single thread which enters into its warp and woof (1979: 95).

Texts appear to have a network structure, but this seems to contradict the conventional view of knowledge. Central to the program that lies behind artificial intelligence is the issue of knowledge representation. The mainstream of Western science has cast mathematics and geometry, in particular, in the paradigmic role. This view holds that our organisation of knowledge is structured in the following fashion: certain elements are held to be foundational and basic, and from these basic axioms, all other components of the system are to be derived. Taking a constructional metaphor, it is like a building where the stones are laid one on top of another, only the foundation having no elements beneath it.

The metaphor of a chain of linked facts is often used to express this idea. How often do we hear it said that an argument is only as strong as its weakest link? But it should not be forgotten that such a model of knowledge is a metaphor, one whose metaphorical nature should not be forgotten. Just as the earlier chapters of this thesis questioned the conversational interface for AI system and proposed an alternative, so too there are alternative ways of conceiving knowledge. The one that is of interest here is the network approach as formulated by Rescher (1979). This network approach sees a system of knowledge as being a set of interrelated theses, not necessarily arranged in a hierarchy (as are axiomatic systems), but connected in an interlacing network.

Just as 'texts' and 'textile' share a common etymological root, so too the idea of weaving arguments together is, in fact, buried deep in our wider vocabulary — it is very common to hear someone complain that they have not 'got the thread of an argument' (Indeed, it is for this reason that the term *thread* is used within Running Texts). 'Implicate' originally meant to twist or fold together, intertwine, entangle; the word 'context' comes from the Latin 'contexere', meaning to weave. More recently Stonier (1984) has also employed the weaving metaphor to account for our creation and use of knowledge. He sees data as being equivalent to a raw fleece, information is like the spun yarn, and knowledge is represented by woven cloth into which patterns of information are brought together.

An advantage of the network approach over an axiomatic system is that certain of the components can be modified without drastic consequences for the whole structure. The foundationalist model requires certain theses to be basic, and these are used to justify other theses. The basic theses are the foundations upon which the structure is built, hence the name foundationalist. But should they be uncertain, as much of our real world knowledge is, the system becomes very shaky. The network model sees the state of our knowledge in a very different way. We have no basic thesis, each statement can be connected to another, and the connections between theses can, if so required, be circular. The validity of a certain knowledge structure is not set by the certitude of any basic theses whereby the correctness of the deduction chain, as would be the case in a foundationalist model, but by the overall strength and coherence of the structure. This is not to say that there are no Euclidian features in the network. Euclidianism is not required globally, but it is quite likely that locally there will be areas of Euclidianism within it. It does seem reasonable that given a text can be composed of a many lines of reasoning (cognitive methods, or *threads* in our terminology). If these various lines of reasoning interconnect and support each other's conclusions, then we can be more content with our state of knowledge. Even if one aspect of our knowledge later proves to be contrary to the discovered facts, our structure will not fall apart totally, although some aspects will have to shift, as a spider's web might if a strand were broken.

Similarly, Spiro (in McGinley & Tierney 1989) was inspired by the later work of Wittgenstein and his use of the metaphor of a 'crisscrossed landscape' for knowledge and learning, in which a domain is treated as a landscape that can be explored from several directions and perspectives. The problem with conventional knowledge representation techniques is that they tend to oversimplify. It is this notion of multiple views that we also try to embody by having any text able to be interpreted by any number of 'ways of reasoning' (for example, rule-based and causal threads to contribute to one argument), which parallel the various paths across a landscape. In the same way that different paths give us differing views of the same landscape to enable us to assemble a complete internal map, the various cognitive methods each contribute one view of the subject domain.

Given this discussion, one might propose that the knowledge medium would not be concerned with the creation of a large axiomatic systems. It would allow for local formal structures, implemented in code, but these would be individual threads in a larger informal structure. The coherence of any 'text' of a knowledge medium lies in the number of threads and their interconnectivity. This argument modifies the original suggestion for a design of a knowledge medium as consisting of a series of AI techniques linked into linear narrative form. Instead of being strictly linear. They can intersect each other, in a supportive fashion.

This brings the whole issue of linearity (a major issue for hypertext) back into the argument. The ConText part of Running Texts is linear, but non-linear, networked, aspects appear elsewhere. The underlying threads might be intersecting and supporting each other in a complex network fashion, to which the reader gains access via the ConText. This upgrading does, as has already been mentioned, create problems for the interface design of the Text Editor. Consider the case where a particular part of the ConText — a lexia — is linked to more than one thread. It might, for example, be the concluding statement from one, and a given fact for another. If the lexia is connected to two threads one can imagine Text Editor design in which there are two gloss panes, one either side of the ConText pane. On a screen small enough for a notebook computer, there would already be problems of space. The solution currently employed is simpler: the author writes separate lexia in the ConText Pane which expresses the link between their two associated glosses (and hence threads).

Typographical Issues : Citation & Indexing

The system described in the previous chapter has its roots not only in AI, but also in typography. Therefore, there are many issues that arise from within typography that a full Running Text system should address. As it is expected that it would be used by 'knowledge workers', those engaged in research and argument, two features of conventional textbooks and research papers that would be required are a means for citation, and indexes to access information in new ways.

Citation is a major concern because of the importance of appropriation (re-authoring) that Running Texts facilitate. How would a re-author (a reader-author) ensure that proper attribution to a previous Running Text was made, allowing for the power of the system to facilitate modification? One approach would be for the re-author to take an existing thread and comment out (between double quotes, which the Smalltalk interpreter treats as comment) parts of the facts or model that they consider wrong or irrelevant. For example, given the causal model of factors affecting the acceptance of novel design, a reader, when appropriating the model for his or her own use, may consider that the link between sex and education is not true for the new target population. Maybe they are all known to be of the same educational level. The reader-author can either just delete the link, or acknowledge the source model as being that of de Bont, Schoormans and Wessel, and paraphrase accordingly: Editing the link in the model to read:

```
"from causal thread: de Bont"
"link (#sex,#education,#neg)  this is not true for this sample- John Smith"
```

The full original thread is kept, so a further reader could see what modifications were made. Academic writers use a similar system. Their original textual source material can either be substantially changed, or it can be quoted directly. Likewise, a thread can be changed drastically, with a reference in the ConText to its author, or it could be extracted and only slightly modified in the form of a 'direct quote'.

New forms of indexes can also be provided for the reader. One possible index can be based on the threads themselves rather than the words in the text alone. Just as writing and print made objects of concepts that could then be located within the space of the book (hence *volume*) through the index, so the Running Text (particularly with an object-oriented implementation) makes an object of a 'line of reasoning'. A Running Text index should be usable to trace an idea back and get a wider view. For instance, on reading thread X in a text A, one could see what other documents use thread X, or modifications of it. Reading the texts associated with these could evoke new ideas for the author on his or her original topic.

One specific new form of index can be seen in the porcelain example above. The reader could be presented with a pane that lists all basic features of porcelain with which the thread deals, and the reader selects recognized features. By selecting , say, three or four features, the reader is lead to an identification and the appropriate ConText. Such a functionality mirrors the usual index, but allows the reader to search through multiple entries. Consider an ordinary book with an index for various features. The reader would then have to note all the page numbers for each of the many features found, and find the page numbers that are common to all. The Running Text makes this process dynamic and automatic. Just as printing and page numbering made the conventional index possible, so the computational nature of the Running Text platform makes this new form of index possible.

Using the gloss and the threads as a means to access a large amount of conventional text does, however, have one precursor in the manuscript world. Such a mechanism has been found to have taken place in Biblical glosses. A less obvious role they performed was that the reader could use the Bible as an index to obtain the opinions of fathers on certain issues.

> The author's [Peter Lombard's] commentaries were arranged, not as in his Sentences by subject matter, but in the *ordo narrationis* of the Bible. The Bible next was therefore extremely useful to all but the most learned scholar since it provided the indicator, as it were, by which the reader found his place in the text (de Hamel 1984: 22).

This feature was of course based on the assumption, reasonable that at the time, that the reader knew his Bible well, and appears to be a uniquely medieval form of index.

This ability for a document to use the task structure, and the readers knowledge of the task, as a means to aid information access have also been implemented in hypertext manuals. The ExpertBook (Simon & Duell 1992) is a structured hypertext, where the links are based on model of the situations that the user would encounter in its domain of nuclear power station maintenance. Facts on the situation as they arrive to the user are employed by the system to infer likely areas of interest. The appropriate pages of the extensive manuals appear on screen. The user's perception of the situation operating over a model of the domain acts a new form of index. Features of the situation guide access to the information rather than a conventional index, where it is a feature of words — their alphabetic ordering — that provides the opportunity for easy access.

Graphic Interfaces

One very significant development will be the use of graphical interfaces to assist the author and reader in the creation and editing of the various models within the threads (For a review of work in using visualisation techniques for knowledge, see Murray & McDaid 1993). For example, the causal reasoning thread would be much aided if the model pane of the Thread Editor had a graphical representation of the causal network, by which one could add or delete nodes and links graphically. Similar work has been undertaken by Eisenstadt et al. (1990) at the Open University. They have implemented a set of graphical tools as parts of the KEATS knowledge engineers workbench. For example, a table representation that automatically creates the rules for a rule-based system. Another knowledge acquisition system, OPAL, used to enter and review the knowledge of oncology treatment plans in the ONCOCIN expert system, has a graphic interface (Musen et al. 1986; Musen, Fagan, & Shortliffe 1986). Knowledge entered graphically into OPAL is automatically converted into the internal format used by ONCOCIN.

The importance of these graphical interfaces is not just to assist the user. They contribute to the argument that we have been following so far. As we have discussed above, lists, tables and diagrams are the product of writing in the widest sense, of pen upon paper. The computer takes these forms and activates them. It is appropriate then that the reader should see the inherent graphic form associated with these structures. In other words these graphical interface types are essentially typographical, they leave the demands of the Turing Test and AI's traditional concerns for conversation exchange.

Interestingly, the idea of using a graphical layout to represent lines of reasoning has also recently been employed in a conventional printed book. The British Medical Journals's *Central Nervous System. Clinical Algorithms* (1989) provides an algorithmic approach to the evaluation and treatment of pain for physicians. Each section consists of a decision tree and related textual comments.

The answer, in its textual form, also has an interesting relationship to the model, when we make the model a graphic or tabular form, such as the Kosy and Wise spreadsheet, for example. As was mentioned in chapter 4, one would not write out verbally the basic table, it is clearer in the spreadsheet form. But the explanation is clearer in the textual form. Equally, although the rail timetable has more clarity overall than its verbal form, for a specific query an explanatory form could be more useful. If, for example, I am standing on the platform and simply want to know when the next train goes and from which platform, the straight answer provided by the timetable suffices. But if I want to know which train at the weekend is going to London, the first answer (that is, the first train going) may be correct but inappropriate. An algorithm similar to that of Kosy and Wise could search the timetable and after giving the first train, move on to (possibly) suggest that the next train arrives earlier in London because of its more direct route, or that travelling the following day would save a great deal of money. Kosy and Wise used a written protocol of what financial analysts actually do to design their algorithm, so too a dynamic timetable could be based on a study of what travellers find important: usually not just the first train out.

These more verbal explanations are only appropriate when a specific query is made, applying it to the whole model would create a mountain of text. But, equally, the list/table/matrix/network or other graphic form seems very appropriate for representing the basic model. There appears to be a specific role for the various formats that the model and the explanations can take, one that addresses a major issue.

8. Effects and Implications

This chapter looks at the more general and long-term implications of viewing artificial intelligence as a form of writing. There are two reasons for exploring expected or potential effects:
- Firstly, as we have seen above, Running Texts, unlike traditional AI systems, require an effort on the part of readers and authors alike. Users will only be prepared to undertake this effort if they can be convinced that there is a long-term benefit that outweighs the effort required. An investigation of writing suggests what these benefits might be.
- Secondly, as well as the benefits, an investigation of the effects of writing may make us aware of any possible negative effects that we should be consider.

Possible effects of an extended 'AI as writing' can be found by investigating the effects that conventional writing and similar notations have had on people after extensive and continual use. That is, we shall use the same strategy as in previous parts of the thesis. If AI and writing share features, it is not unreasonable that aspects of the long-term effects of various forms of writing might apply to AI.

The effort required in learning to read and write a Running Text is crucial. Running Texts differ substantially from other forms of AI program in that they demand much of the reader, and even more of the author, who must understand the various representational languages and how they can be used. However, these demands are ones shared with writing. Unlike spoken language, the learning of writing by children is a major effort, one that many never fully achieve. Historically, too, there are other similarities. Many people in the past could not read or write, and relied upon specialists to do so. This parallels the current situation where a specially trained group of programmers are employed to built systems. And practically, there were problems with writing, not unlike those felt by those coming to computing. Ong (1986) reminds us that in the Middle Ages most people who required something to be written employed the service of a scribe because, in addition to knowing how to write, the scribe had to posses specific skills in scraping and polishing of the animal skin or parchment, whitening it with chalk, sharpening quill pens and mixing inks. Simple pen and paper did not exist.

Likewise, programmers not only have to deal with issues of program content, but also particular technical problems such as standards or networking.

But sharing the problems once associated with writing is unlikely to persuade anyone that they should suffer equally in using a knowledge medium, unless they gain similar benefits to those that that made learning to read and write worthwhile. Exactly what are the benefits from writing?

Implications of Writing

After the influence of McLuhan it is now almost a truism that media impart a systematic bias to the material presented. Recently Havelock (1980, 1982, 1986), Goody (1977, 1986), Eisenstein (1983), Ong (1982) and Stock(1984/5) have made the case that literacy not only extended oral language and memory, but also altered the content and form of knowledge in important ways. The nature of these changes is still disputed, whether they actually affect human cognition, for example, or whether the effects are mediated socially. But that the changes have had significant effects is not in dispute. Most of these effects can be traced back to the concrete nature of writing which makes our thoughts into separate, stable, visible objects.

Before proceeding to discuss the effects of writing, we must not, however, forget that oral communication, although it looks 'natural' and unproblematic, is like writing and computing in many ways (Finnegan 1988). It, too, rests on social and cultural conventions, and there exists a 'technology of memory' to help people remember those things important to their culture. It was necessary to extend everyday memory through the use of verbal and metrical patterns to guarantee the preservation of oral transmission (Havelock 1963, Ong 1982). There was also extensive use of visual memory to remind an orator of what was to be said (Yates 1984).

Externalisation

One important feature of writing, and one that it shares with knowledge-based systems, is that of concretization. By externalising speech onto a surface, it makes concrete what was previously caught up in the transient flow of conversation or thought. The development of writing greatly affected Greek culture through this process of externalisation:

> the alphabet converted the Greek spoken language into an artifact, thereby separating it from its speaker and making it into a *language*, that is, an object available for inspection, reflection, analysis (Havelock 1982: 8).

This visible artifact was preserved without recourse to memory. It could be arranged, reordered, and rethought to create forms of statement and types of discourse not previously available because they were not very memorable. Prior to writing, knowledge had to be maintained in a form that was easy to memorise, giving pre-literate cultural knowledge its ritual, repetitive and poetic form. Although writing did not have the full expressive powers of speech, the capture of many of its features in writing liberated man. The

> imprisonment of spoken language in a system of visual symbols ... did not in fact constitute a real prison but rather provided release, a freedom for language and its resources to expand previous oral standards (Havelock 1982: 47).

One side-effect of the representation of speech as discrete graphic units is that writing tends to represent experience as products, whereas speech represents experience as processes. Compare the similar forms of typical written and spoken forms (Halliday 1989: 81):

Written
Every previous visit had left me with a sense of futility of further action on my part.
Spoken
Whenever I'd visited there before, I'd ended up feeling that it would be futile if I tried to do anything more.

The written form consists more of nouns, of objects, hence *visit, sense, futility, action*. But when you talk you describe what is happening, hence: *had visited, had ended up feeling, tried to do*.

Writing presents a synoptic view of the world, there are things that exist. Speech presents a dynamic view, a world where phenomena do not exist, they happen (Halliday 1989). Writing enables the creation of a conceptual discourse that replaces the poetic one of oral tradition. The fulcrum for this development, according to Havelock (1986) lies in the copula, the commonest use of the word "is". An example of the use of the copula is "your house is beautiful" in which the copula assigns a property to an object. In ancient Greece the "is" would have been omitted. The dynamic flow of daily experience becomes remodelled into a series of "is" statements, that is, the world becomes one where the verb *einai*, 'to be', comes to suppress *kinein*, which meant the disturbance inherent in any real action. Bruner and Olson give an illustration: In oral Greek culture the concept of 'courage'

> is described within temporally ordered narrative accounts of the deeds of the gods, whereas for the literate classical Greeks it is defined in the non-temporal, logical principle "Courage consists in a rational understanding of what is to be feared" (1977/78: 8).

This effect has also been found in other cultures, indication that it is not a result of some particular classical Greek conditions. Bruner & Olson (1977/78) also note that a lack of the copula has been found in unschooled members of the Wolof of West Africa, and a tendency to omit the copula has also been indentified in non-standard dialects of English.

The concretisation of our knowledge that writing constitutes can have powerful effects. Ong (1982) goes so far as to state that literate people are beings whose thought processes grow not only out of their natural powers, but also from those powers that are restructured, either directly or indirectly, though the technology of writing. Writing allows us to establish a context-free discourse, an autonomous discourse which it is not possible to either question or contest, since it has been separated from the author. One result of this separation is the requirement of authors to make their assumptions clearer in the text, as it can be read by people in another time and place. Writing encouraged clearer thought. The increasing trend to what we would call rational thought, possibly through this notion of objectivity, seemed to be closely bound with the extension of writing (Ong 1986). The development of the most formal of languages — logic —

> in the restricted sense of an instrument of analysis procedures ... seemed to be a function of writing, since it was the setting down of speech that enabled man clearly to separate words, to manipulate their order and to develop syllogistic forms of reasoning (Goody 1977: 11).

In the past logic has been taken analogously as being similar either to geometry or to grammar — the analogy goes something like this:

> Both with grammar and with logic we begin with a pre-theoretic corpus of practice; speech in the one case and reasoning in the other. In either case, the theoretical systematisation of the best (best and most correct)

> practice in this area can be made, codifying the procedural principles governing the practice in actual operation. With speech the framework of rules that present the theoretical systematisation of the pre-theoretical practice is grammar, and with reasoning it is logic (Rescher 1977: 242).

The close relationship between what is conventionally considered grammar and the logic systems created by AI has also been observed by some. For example, Briggs (1985) has compared the knowledge representation work of Schank with the work of the Sanskrit grammarians. Within AI much effort has been expended in trying to develop unambiguous representations of natural languages. But similarly, in addition to the creation of many works of literary value, Sanskrit has a long tradition of grammatical studies. Briggs gives the example of the sentence:

> "Out of friendship, Maitra cooks rice for Devadatta in a pot over a fire"

An analysis in the Sanskrit grammarian form would be:

> The Agent is represented by 'Maitra', the Object by 'rice', the Instrument by 'fire', the recipient by 'Devadatta', the Point of Departure (or cause) by 'friendship' (between Maitra and Devadatta), the Locality by 'pot'.

As all these structures are auxiliary to the verb *cook,* we can add it to this analysis, giving a set of triples given on the right hand of the table below. A semantic net representation for this sentence can be written as the set of triples on the left (1985: 37). On the right would be an AI representation of the same sentence in the form used by Schank. The similarity is clear:

cook, agent, Maitra	cook, agent, Maitra
cook, object, rice	cook, object, rice
cook, instrument, fire	cook, instrument, fire
cook, recipient, Devadatta	cook, recipient, Devadatta
cook, because-of, friendship	cause, event, friendship
friendship, Maitra, Devadatta	friendship, object1, Devadatta
	friendship, object2, Maitra
	cause, result, cook
cook, locality, pot	cook, on-loc, pot

All the stranger then that logic should try and forget its written roots, and seek again to become an unmediated 'language of thought'. If we take the analogy of logic as grammar, then the work in AI, based extensively on logic, can be seen in the tradition of employing our writings to refine our thought further — to improve our understanding.

Both writing and AI, then, are technologies in this widest sense. They concern themselves with the creation of artifacts, upon which their creators can reflect, and which can reflect back on their creators. In this regard Clark (1991) has made the observation that, like writing, what computing has to offer are the virtues of being persistent. Facts and records of reasoning, in a systems such as his and Running Texts, do not fade, and the computer's capacity for routine search allows arguments to be thoroughly checked and explored.

Computing stands in a more paradoxical relationship to writing in regard to the suppression of the dynamic for the static, the victory of *einai* over *kinein*. On the one hand the writerly nature of computing is expressed through the growth in object-oriented systems in which class-property relationships are paramount. At the same time the very dynamic nature of computing seems to be returning us to a representation that is more active than passive — it moves. Looking at the examples implemented in the last chapter it is not immediately easy to see whether the threads objectify the text, as they can be copied and moved, or whether their computational nature takes the final text away from such a position.

Writing Engenders Understanding

As well as encouraging the creation of formalism such as logic, the concretisation that writing is also allows us to play with a given text. It is now considered a basic tenet of contemporary theories of writing that it engenders understanding by virtue of the possibility of re-examination that it affords (McGinley & Tierney 1989). This is because when an utterance is put into writing, it can be inspected in much greater detail. Writing fosters the ability to think critically, because of the slow and repeated process of analysis and synthesis, and the ability to review it affords. Gage has said about writing:

> Writing is thinking made tangible, thinking that can be examined because it is on the page and not in the head invisibly floating around. Writing is thinking that can be stopped and tinkered with. It is a way of holding thought still long enough to examine its structure, its flaws. The road to clearer understanding of one's thoughts is travelled on paper. It is through an attempt to find words for ourselves in which to express related ideas that we often discover what we think (in McGinley & Tierney 1989: 248).

Goody (1977) also emphasises this critical aspect of writing. It is easier to recognise contradictions in writing than in the constant flow of conversation, allowing comparison of statements that were made at different times and in different places. When written down, a text can be inspected in much greater detail, in its component parts as well as in its whole, not only forwards but also backwards, out of context as well as in its original setting. It can be subjected to quite a wholly different kind of scrutiny than is possible with a purely verbal communication.

Empirical evidence has been found to support these more general positions. Van Nostrand (1979) reports observations that were documented at the Center for Research in Writing. Analysis was made of some six hundred writing samples produced by forty-five students, each written in response to controlled writing assignments. The assignments were variations on a basic set of instructions in which each writer was presented with the same collection of information fragments. They were then asked to do the following: first state some principle of relationship, then write a paragraph using the information given on this relationship, and finally summarize this paragraph in a sentence. What the experimenters found was that about 85% of the summary sentences incorporated new information beyond that stated in the first principle of relationship. They concluded that "the act of writing engendered new information even within a single paragraph" (1979: 180).

It is not difficult to imagine many reasons why external representations are of assistance. The mind has a limited working memory. Having an external surface to record the steps of a long argument alleviates this problem. But Reisberg (1987) argues that external representations differ from internal, mental, representations in an important way, and one which encourages critical thought. These representations exist after we are no longer thinking about them, and are open to interpretation by others who are not aware of the intent with which it was created. The original author can also interpret the signs in a new way. It is this feature, of re-interpretability, argues Reisberg, that internal representations do not share.

From his work on mental images, Reisberg argues against the position which sees a mental images as picture-like and requiring interpretation by some processes related to visual perception. Instead, he states, they are inherently meaningful, and so unambiguous. He supports this view with empirical work that he and his colleagues have undertaken to test the hypothesis. Subjects were asked to encode some of the classical 'ambiguous figures', such as the duck/rabbit or the Necker cube. The subjects were then asked to form mental images of these figures, and to examine the image to find an alternative meaning for the form. The results were as follows: Absolutely no subjects succeeded in finding alternative construals for their images. However, an experimental control brought out the contrast between the subjects internal and external representations. After the subjects had attempted, and failed to, re-interpret their images, they were asked to make a drawing of the image, and then to try to reconstrue this drawing. All the subjects could do so. Reisberg concludes:

> In imagery, the understanding is inherent in the representation, so that there simply is no representation separate from the understanding. With no freestanding icon to interpret, no reinterpretation is possible (1987: 287).

More significantly, discovery (in terms of seeing things in a new light) is not possible if discovery is considered as a change in an image's understanding. Drawing the image facilitates the discovery process. Reisberg, although his own work is in the area of mental image and drawings, believes the results equally apply to propositional knowledge and texts.

Externalisation, in addition to facilitating critical review, also changes the relation between the the person who writes down the material and that material. Cultures which have never known writing are more integrative, there exists a kind of empathetic identification between knower and known, due to the personal nature of conversation (albeit an identification that Derrida has challenged). With writing this state of affairs undergoes a cleavage, separating the knower from the known universe and then from him or herself. Indeed, it is likely that this cleavage is another feature of writing that encourages and fosters analysis and abstraction. Spoken words differ from written words in that they exist in time and are said within a context, and they encourage a sense of participation in the tasks at hand. So another effect of writing and printing has been an alienation, not in the destructive sense, but one in which writing has 'restructured' consciousness (Ong 1977: 17). One of the best known investigations of the relation between an oral culture and abstract reasoning is the work of Alexander Luria. During the years 1931-2 he gathered data in the two then Soviet provinces of Uzbekistan and Kirghizia and found striking differences between the reasoning of non-literate peasants and those with some degree of literacy (McKnight, Dillon & Richardson 1991).

In one test Luria presented his subjects with drawings of four objects: a hammer, hatchet, log and saw, which they were asked to group into categories. The literate subjects placed the tools into a single category, with the log separate. The non-literate subjects, however, did not use this form of categories — all four objects belonged to one group related to an actual every day task, not components of an intellectual game. As one 25-year-old illiterate put it:

> They're all alike. The saw will saw the log and the hatchet will chop it into small pieces. If one of these has to go, I'd throw out the hatchet. It doesn't do as good a job as the saw (Luria, in McKnight, Dillon & Richardson 1991: 23).

In another test, Luria investigated the capacity to handle formal deductive logic. The non-literate subject exhibited a tendency to reject the self-contained nature of simple syllogisms and turn to their real-life knowledge. Given the syllogism:-

> In the Far North, where there is snow, all bears are white
> Novaya Zembla is in the Far North and there is always snow there.
> What colour are the bears?

A typical reaction of a non-literate was 'I don't know. I've seen a black bear. I've never seen any others... Each locality has its own animals', whereas a semi-literate replied 'To go by your words, they should all be white' (Luria, in McKnight, Dillon & Richardson 1991: 24).

Similar impacts have been found to apply to the extension of writing that is printing. The industrialisation of the word that printing involved led to developments that carry great parallels to a knowledge medium. The impact of printing has been extensively investigated by Elizabeth Eisenstein (1983) who has made many points relevant to the current discussion. For example, highly technical texts, in the days of hand copying, could only be copied by trained minds — just anyone would not do. Thus the introduction of printing released these minds for additional observational and research work. In a similar instance, mathematical tables, the reliable hand copying of which was almost inconceivable on any scale, suddenly became available to any researcher who wanted them — as in the title of a book review by De Solla Price (1967), we can view the 'book as a scientific instrument'. Having a large number of texts in one library encouraged, indeed made imperative, the development and use of classificatory systems such as alphabetic ordering or the Dewey system, which were previously not essential to the sole medieval scholar, who only had a small library to hand. These innovations, however, were not neutral in impact. The invention of titles and indexes marked a change in attitude to text. The book is now felt as a "container in which things are held, not a voice" (Ong, 1977: 88). The innovation lead to a cultural shift. It is from this tradition that he Universal Language Movement came, seeing the world as a collection of objects to be indexed and manipulated.

Printing further enhanced the process that writing began of exposing either contradictions or support for the readers theories and opinions, provided by a variety of texts. It extended the range of texts and hence ideas that could be compared and challenged:

> More abundantly stocked bookshelves obviously increased opportunities to consult and compare different texts...Contradictions became more visible, divergent traditions more difficult to reconcile. The transmission of received opinion could not proceed smoothly once Arabists were set against Galenists or Aristotelians against Ptolemaists. Not only was confidence in old theories weakened, but an enriched reading matter also encouraged the development of new intellectual combinations and permutations (Eisenstein 1983: 43).

Due to these effects, Eisenstein has particularly stressed the role that printing paid in the development of modern science. Indeed, "in some instances, scientific pioneers were themselves printers" (Drake 1970: 47).

AI and User Understanding

The close critical analysis of written material, that has been described by Goody and Havelock, is a feature that knowledge-based systems exhibit, in common with writing. The externalisation of knowledge required for the implementation of knowledge-based systems has been found to assist the clarification of the domain represented. Berry and Broadbent (n.d.) have reported that experts involved in the construction of knowledge-based systems frequently admit to it being a rewarding experience, in that it helps to sharpen up an expert's thinking. Among the first people to consider that knowledge engineering could be used for this purpose were Mark Stefik and Lynn Conway (1982), based on their own experience of designing an expert assistant for VLSI design. In this work may lie the roots of Stefik's later proposal (1986) for the 'Next Knowledge Medium', with which this thesis began.

Unfortunately, because the primary goal of AI is the development of intelligent machines, such effects are rarely documented. But a few cases of the value of the knowledge engineering process, and its potential value, has arisen in the literature. One example of an AI system being built specifically to aid research was that undertaken by Kyle (1986). He built a knowledge base to store results from a research project concerned with the physics of cold winds found in mountain valleys. As in writing, the close, critical, analysis required to create the knowledge base led to the identification of an oversight in the experimental design. Although he reports a large number of problems that he incurred — due to the size of the task — benefits were found. The greatest of these came from the questioning that the knowledge engineer had to undertake. This caused the researchers in the project to refine their thoughts about their results. Specifically, the questioning produced the realisation that infrared cooling of the valley after sunset was the main driving and controlling force of the air flow in the valley. Although Kyle sought the make enhanced understanding the primary project goal, Sergot et al. have sought to use it as a lesser one. In one project they undertook to formalize the British Nationality Act as a logic program, and had, as one of their stated aims that this would "illuminate some of the issues causing the controversy" (1986: 370).

These examples are unusual. In other cases, the ability of knowledge engineering to aid understanding was not a project target, but seemingly a side-effect, which was recognized and subsequently reported. The best documented example of such a situation was Musen et al. (1985). They were concerned with developing a system to provide advice to doctors who treat cancer patients enrolled in clinical trials, and undertook an analysis of the domain knowledge to encode it in the program. The close analysis afforded by this uncovered ambiguities in the original printed forms, which have since been revised. For example, four of the Stanford protocols combined chemotherapy and radiography, but in their written form failed to specify exactly when the radiography treatment should be given. They also failed to mention how long after the radiotherapy one should wait before starting chemotherapy. Only during the process of encoding the protocols for implementation did these issues reveal themselves. They also uncovered vagueness that the original protocol authors could not answer (unlike the previous points which were known but just needed to be made explicit). For example, one protocol stated that one should give 75% of the usual dose of cyclophosphamide if there is renal toxicity, 75% if there is mild myelosuppression, and 75% if chemotheraphy has been previously delayed. There was, however, no indication, and the original authors could not immediately answer this problem when the knowledge engineers raised it, as to what should be done if two or more of these conditions were fulfilled. Is the 75% reduction still acceptable, or 75% of 75%? However, ONCOCIN required specific reductions when multiple drug attenuation rules apply.

In a similar case, Darden (1986) reports the results of Freidland and Kedes (1985), and of Karp (1989). In order to implement a knowledge-based system in the domain of molecular biology they were pushed to develop the abstract concept of the *molecular switch*, since when it has proved of substantial interest to molecular biologists who had previously never considered their material at such an abstract level.

Further cases are less specific about the clearer understanding, but the value of the externalised knowledge in the knowledge-base is indicated by the demand, by other domain experts, for non-executable forms. One of the earliest examples was noted by Feigenbaum. Following work on chemical structure analysis by the DENDRAL expert system team, chemists began writing to Stanford, not asking for copies of the program but for the rules that it contained. These were found to be a much clearer codification of the subject than hitherto (Michie & Johnston 1985). Similarly, Pople's codification of the knowledge base for internal medicine has since been distributed to clinicians in the form of a print-out, after requests (Michie 1983).

More recently, however, the potential of analysis to assist in understanding the problem has been identified by Poulymenakou, Cornford and Whitley (1990). They have proposed using knowledge engineering not for the production of expert systems, but as a means of providing a better understanding of managerial and administrative problem situations. It is to be expected that the number of published results on the way that system construction can aid domain understanding will increase.

Kyle, taking these effects as goals, is interesting in that he also attempted to explore the reasons that lay behind his results. There were benefits, he argued, due to:
- the researchers being forced to make their assumptions explicit. Indeed, this

appeared to be the major cause, and the analysis alone aided reasoning, the system need not be implemented.
-the ability of the system to aid the identification of inconsistencies between various positions taken by the research team.
These two characteristics: of the need to be explicit and the ability to better identify inconsistencies, are also those that were identified above as significant in writing.

An illustration of the program facilitating the identification of improvements to the knowledge-base is found in the newer techniques of 'refining' the knowledge-base with formal methods. In one instance, Politakis & Weiss (1984), have implemented a system which interactively guides the user in improving the rules in a knowledge base. This system, SEEK, suggest options for generalizing or specializing rules, allowing a better match with stored case experience in the domain.

The ability of AI technologies to lead to a modification of the original material, its nature as transformational technology, has naturally lead many (including myself initially) to compare AI to other processing industries — that is, the systems are a knowledge refinery. A similar proposition has been put forward by Alonso-Amo et al. who have created the term 'mindfacturing' to describe the process of creating "explicit, non-expensive, guaranteed, easy-to-use, high-quality knowledge which will be marketed like any other commercial product" (1992: 149-150). They also draw out parallels between a knowledge industry and the iron and steel industry (see Table below).

Tasks of the Knowledge Industry	Knowledge Processing Activities	Resulting Products
Mining	Public and Private Knowledge Acquisition	Establishment of Knowledge Structures: Concepts, Reactions, Function...
Casting	Establishment of: Facts, Heuristics, Concepts & Objects, Events, Associations, Processes, Relations, Statements...	Lists, Tables, Graphs, Rules, Frames, Semantic Networks, Headed Registers, Procedures, Logic Expressions, etc.
Assembly	Knowledge Implementation and Processing	Knowledge Base Expansion and Inference Engine Optimization, Integration with Other Computer Systems
Refining	Refining	Validation and Adaptive Maintenance.

Table 8.1 Comparison between Iron/Steel and a Knowledge Industry (Alonso-Amo 1992: 150)

A proposal for a 'knowledge refinery', whose aims are similar to Alonso-Amo et al., has already been suggested by Michie and Johnston. It was planned as an application of AI technology which

> would take in specialist knowledge in its existing form and debug it, pull it together, carry out creative gap-filling wherever the need becomes evident, and turn out knowledge that is precise, tested and certified correct (1985: 132).

Where I would differ from them is that the notion of certifying the knowledge as being 'correct' seems very ambitious and dubious. Like conventional publishing, the knowledge medium could, and should, be able to allow very differing views of a subject to be represented and distributed. But more importantly, the idea and terminology of 'a transformational technology' should point to writing as a comparable technology to AI, rather than oil refining. What is transformed is not so much the material, as our understanding.

This transformational nature of the work of Michie extends the explicating nature of list, tables and matrices (Goody 1977), and uses this process to promote new understanding. Unlike paper-based structures, however, these are made dynamic, and a consequence is that a clearer view of the data becomes available to the human user. One example, implemented by Niblett and Shapiro (see Michie & Johnston 1985: 123) dealt with the end-game King-Pawn-King. They began with a large, mechanically compiled, *table* that could be used in a blind look-up program. They ran parts of this table through an induction program called ID3, using a semi-automatic procedure. ID3 then created a set of simple decision trees which a human player could easily understand and apply. This whole process can be seen as a dynamic instance of the way that a table written on paper aids abstraction. For example, the predictive and analytic capabilities of the periodic table, as described by Kochen & Lansing (1985) indicates the value of expressing material in a graphic tabular fashion.

Often the results of using programs such as ID3 are described in terms of the machine 'generating new knowledge', but his brings up a range of questions. One problem with the idea that the computer can come up with a new solution is that it must be comprehensible to us humans. The point being, of course, that our comprehension of it implies that the reasons for doing so are latent within human thought. The machine is helping make these reasons available in the externalized form: the media aids thought rather than having autonomous intelligence. It is curious that we have the desire to ascribe intelligence to computers, but not to other media. As Halpern (1990) notes, if a mathematician comes up with an unexpected and valuable result as the consequence of his or her explorations, he or she will be given credit for it. No one would suggest that the blackboard and chalk should share the credit. The medium and notation certainly contributed to the discovery but in a way that media do, not the way that humans do. Similarly, Negrotti (1987) has argued that AI will be completely incapable of generating anything new, but it can, like a telescope, spark of new ideas in the mind of the scientist using the system.

In cases where the machine generates results that are not explicable in human terms, where no restructuring of a human view is achieved through employing the machine, people seem disinterested. For example, when the computer was employed by Appel and Haken to solve the four colour map problem

> there was a curious apathy on the part of pure mathematicians... In giving us the answer without such deeper understanding, by magic, the computer left us feeling that we were both cheated and cheating.... (Halpern 1990: 18).

This is similar to the reaction that expert system users expressed when only an answer was given (see chapter 2).

So the technology can foster the accumulation and restructuring of knowledge, but we cannot fill people's heads. This is why the 'AI as medium' metaphor improves upon the refinery metaphor. As Larsen (n.d.) argues through his own metaphor, although information can be transmitted, knowledge can only be induced. He worries that it is a mistake to think we can fill up the student with knowledge much as we can fill them up with information. He sees the transfer of information as being like the transmission of electricity through wires, where there is a direct physical connection between the two ends of the process. But knowledge is like the induction of electricity in a transformer. With a transformer the flow of electricity is induced from coil to the other, with no direct link. Reading a text facilitates the framing of issues by a self-educating subject, inducing understanding from within his or her own cognitive resources (Angus & Lannemann 1988).

The inductive, transformative, effects on the mind are not only associated with the content of the medium, but also the mechanisms of the medium itself. What one seems to find after extensive use of any 'technology of the intellect' is that features of the technology enter the mind and can encourage its internal resources. This applies even to apparently 'hard' technologies such as the abacus, whose users seem to internalize its operations. Intermediate level users use visible finger movements to accompany mental arithmetic (that is, without the aid of an abacus), and if prevented from moving the fingers, they perform significantly worse in the task (Salomon 1987).

In addition to any psychological changes, cultural effects of AI (in its form as writing) can also be expected. A very interesting investigation into the history and role of a notation has been undertaken by Henrik Sinding-Larsen.

If artificial intelligence techniques offer a new means of representing knowledge then, according to Sinding-Larsen, we can learn much by looking at other historical periods in which a new means of representation was introduced. The specific example that he has explored has been musical notation. The earliest form of written musical notation can be found in *neumes*, in which one can discern the precursor of our contemporary (Western) notation. The neumes were like accents written over the words of a song, indicating when the voice should rise or fall, but giving very inaccurate information about pitch or duration. Much like the Running Text, and writing generally, musical notation grew on the back of a previous practice, in this case song:

> Initially, the notation was associated with the religious plainchant, and only several centuries later was it applied to instrumental music. The monks were mainly concerned with a system for the description and distribution of an *existing vocal tradition* (Sinding-Larsen 1988: 95).

But what lessons can we learn from musical notation? As in most things there were both positive and negative results as the potential of the new notation became clear. On the positive side, as Sinding-Larsen remarks, without such a notation, complex and extraordinary works, such as Stravinsky's *The Rite of Spring*, would be inconceivable. But on the other hand, the previous oral tradition came to be downgraded and even made to change so as to fit the constraints of the new notation.

Sinding-Larsen reports a very specific case-study in the way that music notation introduced difficulties. During the 1880's, the Norwegian folk musician Olav Brenno played an instrument called the *langleik*, a member of the dulcimer family. The pitch of each tone is determined by the position of the frets. After one performance a member of the audience, a tourist, went up to Brenno. Although praising the music, the tourist (who was a classically trained musician) insisted that two of the notes were 'false'. He helpfully offered to correct this by moving the appropriate frets the required few millimetres, to which Brenno agreed. However, the results were disastrous. The classically tempered scale that was imposed was incompatible with Brenno's conception and memory of the tunes he used to play. He finally just gave up playing the instrument. Fortunately, the story has a happy ending. Several decades later, Brenno was contacted by young scholars who had rediscovered the cultural value of the traditional music and the old scales, and they helped him put the frets back to their original positions. He resumed his playing.

This situation was not unusual. In 1850 the Norwegian musicologist L.M.Lindemann wrote:

> The problem of transcribing the melodies does not only consist in the lack of distinction and clarity in the old people's way of singing. Far worse is the fact that one is repeatedly presented with notes that are a quarter of a step higher or lower pitched than the appropriate ones: i.e. notes that are placed exactly in between our half-note steps. It is the task of the collector to determine to which note, the higher or the lower they belong (in Sinding-Larsen 1988: 106).

Lindemann clearly did not recognize that the inadequacy may have been in the tool for the description of the music, rather than in the music itself. In this way we must not think that because our AI representational language cannot represent something, then the failing lies in the thinking that it is trying to represent. There are immense benefits to the representation of knowledge in the machine. We should however, also be aware of the possible pressure from such a knowledge medium to discourage those forms of thinking that are difficult to formalise. In Sinding-Larsen's words: "knowledge that sounds false will slowly be forgotten or actively suppressed" (1988: 107).

These kinds of difficulties have also appeared in the history of writing systems, which have tended toward uniformity, providing codified conventionalized expressions for language (Halliday 1989), and word meanings and spellings becoming standardised. The early Greeks believed that spelling gave a 'correct' picture of pronunciation, a fallacy never challenged in antiquity (Harris 1986). AI has not really had to address these concern yet, due to the (current) lack of any extensive use of its representations. But this may change, because of attempts to standardize existing work on knowledge representation (Neches et al. 1991, Patil et al 1992). One area within this has been the development of an interlingua, or knowledge interchange format (KIF), by which knowledge can be moved from one system to another (Genesereth & Fikes 1992). Ginsberg (1991), however, has questioned the value of such a program. He argues that it would certainly constrain future work in knowledge representation — as musical notation did for Brenno.

9. Conclusion

Summary of the Thesis

This thesis has argued that writing can be seen as a valuable means for understanding and designing symbolic artificial intelligence systems. A broad exploration of the history and practice of writing, hitherto not found in the AI literature, uncovered many similarities that supported the claim, and provided specific issues to aid the understanding and design of AI systems. Applying the new perspective, we can use AI technology to create a medium to communicate our own human intelligence, rather than aiming to create intelligent machines: A new approach in which, as Barrett (paraphrasing Papert's language) expressed very clearly: "the computer is 'an object to think with', not a thinking object" (1988: xxii). A prototype system, called Running Texts, was presented which embodies many of the lessons that could be drawn from the various technologies of the word, from simple ancient writing techniques to hypertext systems.

Contributions of the Thesis

On a practical level, the results in this thesis have involved the integration of both knowledge-based and hypertext techniques. Thus, contributions can be found for its results that are directly applicable to these technologies.

Knowledge-Based Systems

The ideas proposed and explored in this thesis suggest alternative interfaces for existing knowledge-based systems, and expert systems in particular. Treating the machine as a medium of communication between the creator (author) and user (reader) solves a number of problems inherent in the traditional approach. The machine is not given a personality, and so the tendency for the user to anthropomorphize and attribute intelligence to the machine is reduced, if not eliminated. Instead the computational elements of knowledge-based systems are used as components of a montage, creating a dynamic text. Thus, users see the texts produced by the author, with the help of the system, as complex structured documents. The author of the document is in a better situation that the user of a coneventional knowledge-based system for similar reasons. The burden to create an oracle is lessened, and they do not need to rely solely on the formal mechanisms to produce the appropriate answers. Conventional text is added to the formal component, giving the user access to those aspects of the domain knowledge that cannot be formalized.

Hypertext

The work also contributes to the study of hypertexts, showing how they continue developments already present in writing. We question one basic assumption of hypertexts, in which associative non-linearity is seen as a benefit. As an alternative, it is argued that the computer adds dynamic properties to texts, and these dynamic processes increasingly come from AI. The computational properties of the machine are employed within the text, and when the programs are those developed by AI, they can be seen as a means to add structure to the basic text.

Concluding Remarks

This argument, that AI can be usefully seen as a medium, as a form of writing, has already been suggested by others. These proposals have, however, been of a general nature and one that does not provide specific guidelines for the development of working systems. This thesis aims to contribute to the paradigm through a strategy of investigating the more detailed history and consequences of writing in particular. The results provide evidence to support the emerging paradigm, and have also provided guidelines for the design of a prototype knowledge medium.

The premises in the original metaphor, which sees "AI as Writing", have been shown to be too simple. Each of the three terms has required further specification. In terms of 'artificial intelligence', the argument seems to apply to that symbolic form of AI, of which the expert systems of the seventies and eighties are archetypes. The role of connectionist AI systems in any knowledge medium is a topic for future work. The term 'writing' has proved very problematic, currently being used in both its conventional manner and a broader form that should include computing. Even the 'as' has been found to hide a number of dilemmas: one can see it as indicating a metaphorical relationship between the two domains, or (given the broader definition of writing) a sense of class inclusion. It has been argued that these concepts come together in a theoretical approach that has its roots in the work of Lev Vygotsky (1962). We seem to have a virtuous cycle connecting the mind to our external representations, beginning with speech itself. We talk through our problems with others, a conversation that became internalized as inner-speech. Writing re-externalizes this language further, a writing which itself then seems to enter the mind. The computer, too, offers a further intellectual resource that can assist and re-structure our innate cognitive processes, these may then themselves be internalized. There are, accordingly, three main epoches characterized by the externalization of knowledge through each of these means: speech, writing, and information technology (Sinding-Larsen 1987).

I believe that the approach proposed in this thesis shows great fecundity, allowing us to interpret existing AI work, proposing new questions for future research, and providing specific directions for system implementation. But the approach offers other benefits that come an extensive use of 'AI as writing', those benefits that have been identified for conventional writing. If we find a ruined city with written records, we do not (unlike Plato) think that the speakers in the city were destroyed by the documents; on the contrary, we think that their speech was more appreciated, amplified by their writing (Sokolowski 1988). Likewise, the codification of reasoning that AI could become should not mean that we no longer have to think, but that we could think more clearly, and share our thinking more widely. The gradual diffusion of writing into human activity can serve as an historical parallel for the application of artificial intelligence into social exchange. AI can further reform the embodiment of thought that writing was. Just as those who were involved in the early development of writing might find it hard to imagine what Borges and Shakespeare might achieve with it, so we can look forward to immensely creative applications if AI technology is pushed into this direction.

In the short-term, the approach embodied in the Running Text system described also offers benefits for software design. Although to function fully as a medium the author and reader must come to be literate in AI techniques, there is a stepped system of use which the reader can employ. Initially, the reader can, if desired, just read the conventional text component, and only later as they gain interest and confidence move on to first 'read' the formal component in the gloss, and later still modify these to create palimpsests. Equally authors can begin quite simply, using the system to structure a text with one thread type (such as the simple causal thread). Later they can begin to use various thread types and multiple interconnected threads, as they become more proficient. In this regard, the Running Text follows the proposed strategy of 'stepwise structuring' an evolutionary approach to the development of information systems. The basic idea is "to start with a minimally structured system, for example, based on the manipulation of free text by an editor, and later on impose structure as patterns of use evolve" (Sørgaard 1991: 8). If we acknowledge that computing is a form of external representation, in the manner of writing, we can introduce information systems into organizations through systems that allow the creation of known forms of 'writing' such as text and diagrams. Over time authors can add computational structure to these known forms, structure which users will then learn to exploit. In a task-based application this structure reflects the structure of the task-being performed, a strategy that De Young (1990) has employed in the design and implementation of a hypertext systems to support the auditing process, and by Simon (1992) in the domain of power-station maintenance.

An extension of this argument also allows us to integrate the techniques of AI more easily into conventional computing. This is possible by extending the metaphor and viewing computing in general as form of writing. This point has been alluded to already, as in the de Bont et al. example where the AI methods that are implemented could be linked to non-AI threads, in particular threads that represent statistical methods. Indeed, the ideas presented above could applied to fully non-AI applications, for example, to databases. Within an organization, dynamic documents embodying the calls to the database could be sent to various members of staff, who could (in a manner comparable to the case of Running Texts) modify the document to take advantage of (for instance) updates in the database, and create a revised document. Equally significant is the likely increase in the range and availability of pen-based computers (Carr 1991). These systems actually look like a 'tablet', with a stylus to operate the computer through an interface that explicitly draws upon writing: a *notebook user interface* (NUI). As extensive graphic interfaces become available on most machines, the use of dialogue as an interface choice can be expected to decline further.

A New Medium

If one proposes that AI be seen as a medium (Stefik 1986), it follows that it should be possible to employ it in a creative as well as a purely communicative sense. This requires going a step further than the cases discussed in the previous chapters, a step in which AI systems can encourage the natural creative thoughts of those who might use it. There is a potential for an aesthetic use of AI techniques. Many might argue that such a position is inherently unviable, due to the cut-and-paste functionality of the systems described above. In the case of the knowledge medium one might be doubtful of attempts to use such prepackaged lines of reasoning to be uncreative. It seems unreasonable to many that by restructuring existing material we can come up with anything really new. But, as Ong reminds us:

> From More to Shakespeare, adult Tudor authors turned to collection (e.g. Wit's Commonwealth, 1597, or A Treasury Or Storehouse Of Similes) for ideas, phrases, illustrations, and even plots, just as they had done when they were schoolboys. The most resounding and most quoted passages of Shakespeare are generally reworked versions of what anyone could find here. Like Alexander Pope a century later, Shakespeare was less an originator than a consummately expert retooler of thought and expression (in Goody 1977: 117) .

We would not suggest that Shakespeare's failure to reproduce *The Spanish Tragedy* was a failure of memory, when he produced *Hamlet* instead. So, this criticism put aside, what are the characteristics of a knowledge medium that might enable one to consider creative aesthetic uses?

It is ironic that one characteristic of AI systems that were criticized at the beginning of this thesis is one that might possibly be employed creatively. If we look at other media, the ability of the reader to believe temporarily in the characters and the situation is one feature that helps create a fiction. Equally, although the notion that AI can be considered as intelligent is wrong-headed, by which I mean believing that the machine is actually autonomous or intelligent, this feature can be accepted to create a sense of another person and another place. Bolter (1991) points out that artificial intelligence researchers frequently like to see themselves as demi-gods, out to create intelligent and animate machines. They are wrong in the literal use of the term 'creation' as applied to personalities, but right in another. They are demi-gods in the same sense that authors of fiction are, creating characters that live in their own worlds. The willingness of people to impose a human nature on the systems before them could indeed be used for fictive purposes. It is not difficult to imagine an interactive novel where simple AI dialogue techniques could be engaged with the user, who participates in a (fictional) conversation with a character.

Simple forms of such fiction are already appearing. Niesz and Holland (1984) describe one 'reading' of such a program, Robert Lafore's *His Majesty's*

Ship Impetuous. This interactive story employs the kind of simple word matching and responses that characterize ELIZA, but these fictions have their roots as much in adventure games as in AI. The adventure game nature of these fictions causes a problem as they are close-ended fictions. That is, the reader must find the right 'key' to solve the puzzle, failure at any stage en-route often leading to the reader's death (in the work). The reader must restart and, through trial and error, arrive at the 'correct' solution. A more exciting possibility would be to let such systems into the hands of a writer who could explore the metafictional qualities, where the system leads to tales within tales. One can only imagine what an author like Calvino would make of such possibilities.

Another important aspect that allows creativity is potential in the 'play of the signifiers'. In such a situation the form of the symbol system rather than its 'meaning' is responsible for carrying the ideas forward. Such plays have even been found in Sumerian writing. Acoustic ambiguities were not easily represented in writing, but formed an essential component of creative writing in Sumeria; new and esoteric word games based on the visual character of the script developed, such as acrostics, visual puns and parodies of literary genres (Green 1981). An example of the means by which the structure of the media can create new material comes in the program FRUMP from Yale University. This program aims to summarize news items that come over the press wire. There is a story that one day an item arrived over the wire which began: "Rome was shaken this morning to learn of the death of Pope John Paul I..." The computer system gave out a summary that an earthquake in Rome had resulted in one casualty (Bolter 1991: 180). Although AI researchers would see this as a failure it can also been seen as a creative product of the structure of the medium — a poetics of the representation, one that could contribute to an interactive fiction. The very notion of failure can be seen in a new light, if we look to critical theory. As Eagleton (1983) notes, Heidigger has used the hammer as a model of the knowable object. But we usually take such things for granted until they fail to function, thus a broken hammer is suddenly more of a hammer than an unbroken one. The hammer becomes 'foregrounded', which for the Formalists is a crucial feature of literature. That is, when we read literature we are more aware of language when the language that confronts us appears different to that which we meet in everyday circumstances (Sturrock 1986). In the FRUMP example, it is a failure (in traditional AI terms) of AI, its unpredictability and tendency to brittleness, that could also be used to contribute to its creative potential.

Taking FRUMP in this spirit means rejecting the conventional demand of an AI system, in which it must create the 'correct' answer. Such systems are generative, and can produce a variety of unexpected outputs. Post-modernist writers in particular have often used the computer as a metaphor for the generative properties of fiction, becuase of the way that both create alternatives. Italo Calvino, for example, remarks that:

> The literature machine can perform all the permutations possible on a given material, but the poetic result will be the particular effect of one of these permutations on a man endowed with a consciousness and an unconsciousness, that is, an empirical and historical man (in Stutt 1990: 71).

That is, he machine is good at generation of alternatives, many of which may be absurd. Final selection from among possibilities is a task better achieved humans who have a cultural position from which to address their fellows. Stutt has noted the various attempts to get computers to create poetry and short tales, and their failure, which is due to the fact that they are ahistorical. He thus proposes to apply the lessons of Calvino:

> *Calvino's Principle*: Since computers are good at the combination, permutation and matching of symbols but not at judging the results of these operations, designers of computer systems should attempt to produce systems that assist in the task of forming judgements rather than systems which make final judgements (Stutt 1990: 73).

The knowledge media can fulfill these requirements, embodying permutations but leaving the interpretation to the reader — "the computer serves as a medium of discourse in which different commitments and their consequences can be jointly explored" (Winograd 1990: 188).

The Medium in The Mind of Man

This thesis began with Stefik's proposal that AI should take up a new goal, the creation of an expressive medium rather than the creation of an artificial intelligence. Many of the issues that have arisen in this investigation, however, lead us to a possible long-term possibility — one which asks us to look more closely at the meaning of the phrase 'artificial intelligence'. As has been discussed, there seems to be an intimate link between the mind of man and woman, and the cognitive tools that he or she uses. The potential role that the knowledge medium can have, as opposed with the traditional approach to AI, can be summed up by Ong's statement that:

> Writing and printing and the computer enable the mind to constitute within itself - not just on the inscribed surface or on computer programs - new ways of thinking, previously inconceivable questions, and new ways of searching for responses...The mind does not enter into the alphabet or print or the computer so much as the alphabet or print or the computer enters the mind (1977: 46-47).

The ability for writing to enter the mind also suggests a reason why the idea of artificial intelligence as a simulation of cognitive behaviour is perhaps not contradictory to the ideas in this thesis. If the mind has integrated within itself aspects of writing, then the computational activation of these forms might be both an extension of writing into a new form, and a 'simulation' of the writerly aspects of cognition, at one and the same time.

Agre recognized this situation, observing that almost all the domains that the very earliest work in AI, undertaken by Newell and Simon, where in which an agent interacts with a world that is almost entirely a sheet of paper. Significantly, "Newell's production system models do not contain separate mechanisms for the scratch paper and for the agent's 'short-term memory" (1988: 13). When we internalize the mechanisms of the written word, we can activate it employing components of our natural intellect. The new computational writing tools and the simulations of the cognitive aspects of internalized writing thus become one.

If Ong were right in his notion of a 'transformative technology', one that transform the mind, it would offer the possibility that a truly communicative knowledge medium would not just communicate, but modify the mind. It is an idea that has gained widespread currency. Jerome Bruner has made use of the concept of a 'cultural amplifier'. in which:

> Man is seen to grow by the process of internalizing the ways of acting, imaging, and symbolizing that 'exist' in his culture, ways that amplify his powers. He then develops these powers in a fashion that reflects the uses to which he puts [them] (Bruner, in Cole & Griffin 1980: 344).

Bruner himself drew extensively on the ideas of Weston La Barre, including his contention that human nature in the last half million years has been a result of the human capacity to incorporate external aspects of the world, a process he called 'evolution-by-prosthesis'.

This ability of a medium, a technology of the intellect, to enter into the mind has been uncovered in empirical studies. For example, Salomon's

(1987) reports on abacus users described in the previous chapter. From such evidence, Salomon argues strongly that a symbol system, such as film, can cultivate mental abilities. One of Solomon's students reported the following:

> I am fascinated by how my daydreaming is influenced by movies. Processes and techniques of presenting events by the Hollywood symbol system are powerfully implanted within my cognitive system. I have observed third person narration, flashbacks, zooms, slow-motion emphasis of action, audience viewing, re-takes, 'voice of conscience', multipersonality dialogue, background music, and many other movie means of expression in my head. I fear that there is very little original style to my daydreaming. It is all influenced by celluloid (1987: 262).

There are unique aspects to each media. Writing, print, film and ballet say different things because of the symbol systems within which they operate. Technology extends the ability to say things with these symbols; for example, the way that copper engraving made finer detail possible, the zoom lens extended the powers of photography. The computer is equally a symbol extensor. Her studies lead Salomon to believe that a "media's symbol systems have their own important effects on people's minds" (1987: 253). A knowledge medium would have its own effects, presumably on reasoning and judgment. Should knowledge-based media have such an effect, we are left with the very strange possibility that an artificial intelligence is possible, and soon. But *this* artificial intelligence would not be an intelligent machine, but ourselves — transformed by extensive use of the techniques of AI to express and communicate our ideas.

Bibliography

Agre, P. E. (1988) "Writing and Representation,"*Working Paper 315*, MIT, AI Laboratory.

Akscyn, R. M., D. L. McCracken & E. A. Yoder (1988) "KMS: A Distributed Hypermedia System for Managing Knowledge in Organizations," *Comm. of the ACM* 31 (7): 820-35.

Allen, L. (1986) "The Cost of an Expert," *Computerworld*, July 21: 59-68.

Alonso-Amo, F., et al. (1992) "From Epistemology to Gnoseology: Foundations of the Knowledge Industry," *AI & Society* 6: 140-165.

Anderson, P. B. & B. Holmqvist (1990) "Interactive Fiction. Artificial Intelligence as a Mode of Sign Production," *VENUS Report No.2*, Department of Information and Media Science, University of Aarhus, Niels Juelsgade 84, DK-8200, Aarhus N, Denmark.

Angus, I. H. & J. W. Lannemann (1988) "Questioning the Institutional Boundaries of the US Communications Research: An Epistemological Inquiry," *Journal of Communication* 30 (3): 62-74.

Applebee, A.N. (1984) "Writing and Reasoning," *Review of Educational Research* 54 (4): 577-596.

Barthes, R. (1975) *S/Z*. London: Jonathan Cape.

Barr, A. & E. A. Feigenbaum (1981/2) *The Handbook of Artificial Intelligence: Vol 1 & 2*. Los Altos: William Kaufmann.

Barrett, E. (1988) "Introduction: A New Paradigm for Writing with and for the Computer," Barrett, E. (ed.) xiii-xxv.

Barrett, E., ed. (1988) *Text, Context, and HyperText: Writing for and with the Computer*. Cambridge, Mass.: MIT Press.

Barret, E., ed. (1989) *The Society of Text*. Cambridge, Mass.: MIT Press.

Basden, A. (1983) "On the Application of Expert Systems," *Int. J. Man-Machine Studies* 19: 461-177.

Baumann, G., ed. (1986) *The Written Word: Literacy in Transition*. Oxford: Clarendon Press.

Berman, B. J. (1992) "Artificial Intelligence and the Ideology of Capitalist Reconstruction," *AI & Society* 6: 103-114.

Bernstein, B. (n.d.) "Euclid: Supporting Collaborative Argumentation with Hypertext," *Report*. Dept. of Comp. Sci., University of Colorado, Boulder, Colorado 80309-0430.

Bernstein, B., P. Smolensky & B. Bell (1989) "Design of a Constraint-Based Hypertext System to Augment Human Reasoning," *Report No. CU-CS-423-89*, Department of Computer Science, University of Colorado, Boulder, CO.

Berry, D. C. & D. E. Broadbent (n.d.) "Expert Systems and the Man-Machine Interface," Unpublished Report. Dept. of Experimental Psychology, University of Oxford.

Bever, T. G. et al. (1990) "Spacing Printed Text to Isolate Major Phrases Improves Readability," *Visible Language* 25 (1): 74-87.

Bledsoe, W. (1986) "I Had a Dream: AAAI Presidential Address, 12 August 1985," *AI Magazine* 7 (1): 57-61.

Boden M. (1987) *Artificial Intelligence and Natural Man*. London: MIT Press, revised edition.

Boden, M., ed. (1990). *The Philosophy of Artificial Intelligence*. New York: OUP.

Bolter, J. D. (1989) "Beyond Word Processing: The Computer as a New Writing Space," *Language and Communication* 9 (2/3): 129-142.

Bolter, J. D. (1991) *Writing Space: The Computer, Hyperext, and the History of Writing*. Hillsdale, NJ: Lawrence Erlbaum Associates.

Bouwman, M. J. (1983) "Human Diagnostic Reasoning by Computer," *Management Science* 29 (6): 653-72.

Boyd-Barret, O. & P. Braham, eds. (1987) *Media, Knowledge and Power*. London: Croom Helm.

Breuker, J. et al. (n.d.) "Model-Driven Knowledge Acquisition: Interpretation Models," *Memo 87: Esprit Project 1098*, Dept. of Social Science Informatics, University of Amsterdam, Herengracht 196, 1016 Amsterdam.

Briggs, R. (1985) "Knowledge Representation in Sanskrit and Artificial Intelligence," *AI Magazine* 6 (1): 32-39.

British Medical Journal (1989) *Central Nervous System. Clinical Algorithms*. Cambridge: CUP.

Brown, P. J. (1988) "Linking and Searching within Hypertext," *Electronic Publishing* 1 (1): 45-53.

Brown, P. J. (1989) "Do We Need Maps to Navigate Round Hypertext Documents?" *Electronic Publishing* 2 (2): 91-100.

Bruffee, K. A. (1984) "Collaborative Learning and the 'Conversation of Mankind'," *College English* 46 (7): 635-52.

Bruner, J. S. & D. R. Olson (1977/78) "Symbols and Texts as Tools of the Intellect," *Interchange* 8 (4): 1-15.

Burn, A. R. (1983) *The Pelican History of Greece.* Harmondsworth, Middlesex: Penguin.

Bylander, T. & B. Chandrasekaran (1986) "Generic Tasks for Knowledge Based Reasoning: The 'Right' Level of Abstraction for Knowledge Acquisition," *Technical Report*, Laboratory for Artificial Intelligence Research, Dept. of Computer & Information Science, Ohio State University, Columbus, Ohio 43210, April 1986.

Cajori, F. (1928/74) *A History of Mathematical Notations.* 2 Vols. La Salle, IL: Open Court.

Calvino, Italo. (1992) *If on a winter's night a traveller.* London: Minerva.

Carlson, D. A. & S. Ram (1990) "HyperIntelligence: The Next Frontier," *Communications of the ACM* 33 (3): 311-21.

Carr, R. M. (1991) "The Point of the Pen," *Byte*, February, 211-21.

Cavalier, T. & R. Chandhok (1991) "Graphic Design for a Collaborative Workstation: Columns for Commenting and Annotation," *Information Design Journal* 6 (3): 187-198.

Chappell, D. & C. J. Willis (1992) *The Architect in Practice.* Oxford: Blackwell Scientific.

Churcher, P. R. (1989) "A Common Notation for Knowledge Representation, Cognitive Models, Learning and Hypertext," *Hypermedia* 1 (3): 235-54.

Clancey, W. J. (1985a) "Heuristic Classification," *Artificial Intelligence* 27: 289-350.

Clancey, W. J. (1985b) "Acquiring, Representing, and Evaluating A Competance Model of Diagnostic Strategy," *Report No. STAN-CS-85-1067*, Dept. of Comp. Sci., Stanford University, CA.

Clanchy, M. T. (1979) *From Memory to Written Record.* London: Edward Arnold.

Clark, P. A. (1991) "Model of Argumentation and Its Application in a Cooperative Expert System." PhD Thesis, Turing Institute, University of Strathclyde, Glasgow.

Cole, M. & P. Griffin (1980) "Cultural Amplifiers Reconsidered," Ed. Olson, D. 343-364.

Collins, H. M., R. H. Green & R. C. Draper (1985) "Where's the Expertise?: Expert Systems as a Medium of Knowledge Transfer," Proc. *Expert Systems '85.* British Computer Society SIGES, December 1985: n.p.

Conklin, J. (1987) "Hypertext: An Introduction and Survey," *Computer* 9: 17-41.

Conklin, J. & M. Begeman (1988a) "gIBIS: A Hypertext Tool for Exploratory Policy Discussion," *ACM Trans. on Office Information Systems* 6 (4): 303-31.

Conklin, J. & M. Begeman (1988b) "gIBIS: A Tool for all Reasons," *Technical Report Number STP-252-88*, Microelectronics and Computer Technology Corporation, 3500 West Balcones Centre Drive, Austin, Texas 78759.

Conklin, E. J. & K. C. B. Yakemovic (1991) "A Process-Oriented Approach to Design Rational," *Human-Computer Interaction* 6: 357-391.

Coombs, M. & J. Alty (1984) "Expert Systems: An Alternative Paradigm," *International Journal of Man-Machine Studies* 20 (1): 21-43.

Coulmas, F. (1989) *The Writing Systems of the World.* Oxford: Basil Blackwell.

Cyrus, V. J. (1971) "Linguistic Features of Scribal Spacing," *Visible Language* 5 (2): 101-110.

Darden, L. (1986) "Viewing the History of Science As Compiled Hindsight," *Computer Science Technical Report Series No. UMIACS-TR-86-20 or CS-TR-1715*, University of Maryland.

Davies, W. V. (1990) "Egyptian Hieroglyphs", British Museum (ed.) *Reading the Past.* London: British Museum Publications.

de Bont, C. J. P. M., J. P. L. Schoormans & M. T. T. Wessel (1992) "Consumer Personality and the Acceptance of Product Design," *Design Studies* 13 (2): 200-8.

de Hamel C. F. R. (1984) *Glossed Books of the Bible and the Origins of the Paris Booktrade.* Woodbridge, Suffolk: Brewer.

Derrida, J. (1967/76) *Of Grammatology.* Trans. Gayatri Chakravorty Spivak. Baltimore: John Hopkins University Press.

De Young, L. (1990) "Linking Considered Harmful," Eds. Streitz, Rizk & Andre 238-49.

De Solla Price, D. (1967) "The Book as a Scientific Instrument," *Science* 58: 102-4.

Digitalk (1991) *Smalltalk/V Mac Goodies Tools and Applications.* Digitalk, 9841 Airport Boulevard, Los Angeles, California 90045.

Drake, S. (1970) "Early Science and the Printed Book: The Spread of Science Beyond the Universities," *Renaissance and Reformation* 6 (3): 43-52.

Eagleton, T. (1983) *Literary Theory.* Oxford: Basil Blackwell.

Ehlich, K. (1983) "Development of Writing as Social Problem Solving," Eds. Coulmas, F. & K.d Ehlich *Writing in Focus* Berlin: Mouton, 99-129.

Ellis, C., ed. (1989) *Expert Knowledge and Explanation: The Knowledge-Language Interface.* Chichester: Ellis Horwood

Enos, R. L., ed. (1990) *Oral and Written Communication. Historical Approaches.* Sage Publications: Newbury Park, CA.

Eisenstadt, M. et al. (1990) "Visual Knowledge Engineering," *IEEE Transactions on Software Engineering* 16 (10): 1164-77.

Eisenstein, E. (1983) *The Printing Revolution in Early Modern Europe.* Cambridge: CUP.

Ericsson, A. K. & H. A. Simon (1984) *Protocol Analysis.* Cambridge, Mass: The MIT Press

Feigenbaum, E. A. & P. McCorduck (1984) *The Fifth Generation.* London: Michael Joseph.

Finnegan, R. (1988) *Literacy and Orality: Studies in the Technology of Communication.* Oxford: Basil Blackwell.

Fischer, G. et al. (1991) "Making Argumentation Serve Design," *Human-Computer Interaction* 6: 393-419.

Fischer, G. & R. Christian (1988) "Knowledge-Based Spreadsheets," *Proc. AAAI-8*, St. Paul, Minnesota, Aug 21-26: 802-7.

Fischer, G., R. McCall & A. Mørch (1989) "JANUS: Integrating Hypertext with a Knowledge-Based Design Environment," *Proc. Hypertext '89*, 105-17.

Forsyth, R., ed. (1984) *Expert Systems: principles and case studies.* London: Chapman and Hall.

Friedland, P. & L. H. Kedes (1985) "Discovering the Secrets of DNA," *Comm. of the ACM* 28 (11): 1164-86.

Frisse, M. E. & S. B. Cousins (1989) "Information Retrieval From Hypertext: Update on the Dynamic Medical Handbook Project," *Proc. Hypertext '89* 199-212.

Galegher J., R. E. Kraut & C. Egido, eds. (1990) *Intellectual Teamwork: Social and Technological Foundations of Cooperative Work.* Hillsdale, NJ: Lawrence Erlbaum.

Gaur, A. (1987) *A History of Writing.* London: The British Library, revised edition.

Gelb, I. J. (1963) *The Study of Writing.* London: University of Chicago Press.

Genesereth, M. R. & R. E. Fikes (1992) Knowledge Interchange Format Version 3.0 Reference Manual. *Report Logic-92-1,* Comp. Sci. Dept., Stanford University.

Ginsberg, M. L. (1991) "Knowledge Interchange Format: The KIF of Death," *AI Magazine*: 57-63.

Göranzon, B. & I. Josefson, eds. (1988) *Knowledge, Skill and Artificial Intelligence.* London: Springer-Verlag.

Goody, J. (1977) *The Domestication of the Savage Mind.* Cambridge: CUP.

Goody, J. (1986) *The Logic of Writing and the Organisation of Society.* Cambridge: CUP.

Goody, J. (1987) *The Interface between the Written and the Oral.* Cambridge: CUP.

Graubard, S. R., ed. (1988) *The Artificial Intelligence Debate. False Starts, Real Foundations.* Cambridge, Mass.: MIT Press.

Gray, G. L. (1992) "Combining Expert Systems and Hypertext: Opportunities and Obstacles," *Intelligent Systems in Accounting, Finance and Management* 1: 21-28.

Green, M.W. (1981) "The Contruction and Implementation of the Cuneiform Writing System," *Visible Language* 15 (4): 345-72.

Guindon, R., ed. (1988) *Cognitive Science And Its Applications For Human-Computer Interaction.* Hillsdale, New Jersey: Lawrence Erlbaum.

Haarmann, H. (1990) *Universalgeschichte der Schrift.* Frankfurt: Campus Verlag.

Hacking, I. (1983) *Representing and Intervening.* Cambridge: CUP.

Halasz, F. G., T. P. Moran & R. H. Trigg (1987) "NoteCards in a Nutshell," *Proc. CHI+GI 1987* : 45-52.

Halliday, M.A.K. (1989) *Spoken and Written Language.* Oxford: OUP.

Halpern, M. (1990) *Binding Time - Six Studies in Programming Technology and Milieu.* Norwood, NJ: Ablex Publishing.

Harris, R. (1986) *The Origin of Writing.* London: Duckworth.

Harnad, S. (1990) "The Symbol Grounding Problem," *Physica* D 42: 335-46.

Havelock, E. A. (1980) "The Coming of Literate Communication to Western Culture," *Journal of Communication* 30: 90-98.

Havelock, E.A. (1982) *The Literate Revolution in Greece and its Cultural Consequences.* Princeton, NJ: Princeton University Press.

Havelock, E.A. (1986) "The Alphabetic Mind: A Gift of Greece to the Modern World," *Oral Tradition* 1 (1): 134-50.

Hayes, P. & J. Pepper (1989) "Towards An Integrated Maintenance Advisor," *Proc. Hypertext '89* : 119-27.

Hayes-Roth, F. (1985) "Rule-Based Systems," *Comm. of the ACM* 28 (9): 921-32.

Hill, W. C. (1989) "The Mind at AI: Horseless Carriage to Clock," *AI Magazine*, Summer 1989, 29-41.

Hodgkin, A. (1986) "New Technologies in Printing and Publishing: The Present of the Written Word," Ed. Baumann, G. 151-68.

Hofmann, M., U. Schreiweis & H. Langendoerfer (1990) "An Integrated Approach to Knowledge Acquisition by the Hypertext System CONCORDE," Eds. Streitz, Rizk & Andre 166-79.

Holland, N. N. (1988) *The Brain of Robert Frost.* New York: Routledge.

Hutchins, E. (1990) "The Technology of Team Navigation," Eds. Galegher J., R.E. Kraut & C. Egido 191-220.

Hutchins, W.J. (1986) *Machine Translation: Past, Present, Future.* Chichester: Ellis Horwood.

Huuskonen, P. (1992) "Explaining Diagnosis Through Contexted Hypertext," *Proc. Informatique '92, Interface to Real & Virtual Worlds*, Montpellier, March 23-27: 347-56.

Huxor, A. P. (1986) "Artificial Intelligence as the Basis for a Knowledge Industry," *Proc. First Australian AI Congress*, Melbourne, November 1986

Huxor, A. (1988) "Explanation and Understanding: An Architecture for Economic Elaboration," *Technical Report TR-LP-26*, 27 January 1988, ECRC, Arabellastrasse 17, 8000 Munich 81, FRG

Jamison, D. L. (1991) "Michel Meyer's Philosophy of Problematology: Toward a New Theory of Argument," *Argumentation* 5 (1): 57-68.

Jaynes, J. T. (1989) "Limited Freedom: Linear Reflections on Nonlinear Texts," Ed. Barret, E. 148-61.

Jewett, D. L. (1981) "Multi-level Writing in Theory and Practice," *Visible Language* 15 (1): 32-40.

Karp, P. D. (1989) "Hypothesis Formation and Qualitative Reasoning in Molecular Biology," PhD Thesis, Dept. of Comp. Sci., Stanford University, Stanford, CA.

Knowlson, J. R. (1975) *Universal language schemes in England and France 1600-1800*. Toronto: University of Toronto Press.

Knowlson, J. R. (1963) "The Ideal Languages of Veiras, Foigny, and Tyssot de Patot," *Journal of the History of Ideas* 24: 269-78.

Koch, C. (1992) "Combining Connectionist and Hypertext Techniques in the Study of Texts: A HyperNet Approach to Literary Scholarship," *Literary and Linguistic Computing* 7 (4): 209-17.

Kochen, M. & J. Lansing (1985) "On Maps for Discovery: Did the Periodic Table Guide Elemental Discovery?" *Scientometrics* 7(3-6): 327-39.

Kornell, J. (1987) "Formal Thought and Narrative Thought in Knowledge Acquisition," *Int. J. Man-Machine Studies* 26: 203-12.

Kosy, D. W. & B. P. Wise (1984) "Self-Explanatory Financial Planning Models," *Proc. AAAI '84*, 176-81.

Kuipers, B. & J. P. Kassirer (1984) "Causal Reasoning in Medicine: Analysis of a Protocol," *Cognitive Science* 8: 363-85.

Kurzweil, R. (1990) *The Age of the Intelligent Machine*. Cambridge, Mass.: MIT Press.

Kyle, T. G. (1986) "Using AI as a Research Tool," *SIGART Newsletter* 97: 29-31.

Lakoff, R. T. (1982) "Some of my Favorite Writers are Literate: The Mingling of Oral and Literate Strategies in Written Communication," Ed. Tannen, D. 239-60.

Langley, P. et al. (1987) *Scientific Discovery: Computational Explorations of the Creative Process*. Cambridge, Mass: MIT Press.

Langlotz, C. P. & E. H. Shortliffe (1989) "The Critiquing Approach to Automated Advice and Explanation: Rationale and Examples," Ed. Ellis, C. 179-200.

Larsen, S. (n.d.) "Information can be Transmitted but Knowledge must be Induced," *PLET* 23 (4), Overseas Issue: 331-36.

Leith, Philip (1990) *Formalism in AI and Computer Science*. Ellis Horwood.

Lenat, D., M. Prakash & M. Shepherd (1986) "CYC: Using Common Sense Knowledge to Overcome Brittleness and Knowledge Aquisition Bottlenecks," *AI Magazine* 6 (4): 65-85.

Linster, M. & M. A. Musen (1992) "Use of KADS to Create a Conceptual Model of the ONCOCIN Task," *Knowledge Acquisition* 4: 55-87.

Lipscombe, B. (1989) "Expert Systems and Computer-Controlled Decision making in Medicine," *AI and Society* 3: 184-97.

Manuel, T. (1986) "What's Holding Back Expert Systems?" *Electronics* August 7: 59-63.

March, K. S. (n.d.) "Weaving, Writing and Gender," *Man* 18: 729-44.

Marshall, C. C. (1989) "Representing the Structure of a Legal Argument," *Technical Report SSL-89-30*, Xerox PARC, Palo Alto, CA.

Marshall, C. C. & P. M. Irish (1989) "Guided Tours and On-Line Presentations: How Authors Make Existing Hypertext Intelligable for Readers," *Proc. Hypertext '89* : 15-26.

McCall, R. J. et al. (1990) "PHIDIAS: Integrating CAD Graphics into Dynamic Hypertext," Eds. Streitz, Rizk & Andre 154-65.

McCorduck, P. (1988) "Artificial Intelligence: An Aperçu," Ed. Graubard, S. R. 65-83.

McGinley, W. & R. J. Tierney (1989) "Traversing the Topical Landscape: Reading and Writing as Ways of Knowing," *Written Communication* 6 (3): 243-69.

McKnight, C., A. Dillon & J. Richardson (1991) *Hypertext in Context*, Cambridge: CUP.

Meyer, M. (1983) *Meaning and Reading*. Pragmatics & Beyond IV: 3, Amsterdam: John Benjamins.

Miall, D. S. (1990) *Humanities and the Computer: New Directions*. Oxford: Clarendon Press.

Michie, D. (1983) "Inductive Rule Generation in the Context of the Fifth Generation," SERC & Dept. of Industry, *Intelligent Knowledge-Based System: a programme for action for the UK.*

Michie, D. & R. Johnston (1985) *The Creative Computer.* Harmondsworth, Middlesex: Pelican Books.

Miller, M. & J. Miller eds. (1991) *Miller's Antiques Checklist Porcelain.* London: Mitchell Beazley Publishers.

Milne, Rt. (1990) "Where is the Business Benefit?" *Proc. of Business Benefits of Expert Systems Conf. (Application Stream),* 18-20 September, London, UK: 1-7.

Mitchell, W. J. (1990) *The Logic of Architecture.* Cambridge, Mass: MIT Press.

Monterege, J. T. (1989) "The Creation of Digital Consciousness," *SIGART Newsletter* 109: 30-33.

Motta, E., T. Rajan & M. Eisenstadt (1990) "Knowledge Refinement as a Process of Model Refinement," *Knowledge Aquisition* 2: 21-49.

Murray, B. S. & E. McDaid (1993) "Visualizing and Representing Knowledge for the End User: A Review," *International Journal of Man-Machine Studies* 38: 23-49.

Musen, M. A. et al. (1985) "Knowledge Engineering for a Clinical Trial Advice System: Uncovering Errors in Protocol Specification," *Report Number KSL-85-51* Medical Computer Science Group, Knowledge Systems Laboratory, Stanford University, Stanford, CA.

Musen, M. A. et al. (1986) "OPAL: Toward the Computer-Aided Design of Oncology Advice Systems," *Proc. of Computer Applications in Medical Care,* October 25th-26th, Washington DC : 43-51.

Musen, M. A., L. M. Fagan & E. H. Shortliffe (1986) "Graphical Specification of Procedural Knowledge for an Expert System," *Proc. Computer Society Workshop on Visual Languages,* June 25-27, Dallas, Texas, 167-78.

Nebel, B., Rich, C., & Swartout, W. (1992) *Principles of Knowledge Representation and Reasoning: Proc. of the Third International Conference.* San Mateo, CA: Morgan Kaufmann.

Neches, R. et al. (1991) Enabling Technology for Knowledge Sharing. *AI Magazine* 12(3): 36-55.

Negrotti, M. (1987) "The Piping Of Thought and the Need for a Permanent Monitoring of the Cultural Effects of Artificial Intelligence," *AI & Society* 1: 85-91.

Newman, S. E. & C. C. Marshall (1992) "Pushing Toulmin Too Far: Learning From an Argument Representation Scheme," *Technical Report, SSL-92-45*, Xerox PARC, Palo Alto, CA.

Neuwirth, C. M. & D. S. Kaufer (1989) "The Role of External Representations in the Writing Process: Implications for the Design of Hypertext-based Writing Tools," *Proc. Hypertext '89* : 319-41.

Niesz, A. J. & N. N. Holland. (1984) "Interactive Fiction," *Critical Inquiry* 11: 110-29.

Norris, C. (1982) *Deconstruction: Theory and Practice*. London: Methuen.

Norris, C. (1987) *Derrida*. London: Fontana.

Östberg, O. (1988) "Applying Expert Systems Technology: Division of Labour and Division of Knowledge," Eds. Göranzon, B. & I. Josefson 169-83.

Olson, D. R. (1980) *The Social Foundations of Language and Thought: Essays in Honor of Jerome S. Bruner*. New York: W. W. Norton.

Ong, W. J. (1977) *Interfaces of the Word*. Cornell University Press.

Ong, W. J. (1982) *Orality and Literacy*. London: Methuen.

Ong, W. J. (1986) "Writing is a Technology that Restructures Thought," Ed. Baumann, G. 23-50.

Partridge D. & Y. Wilks, Eds. (1990) *The Foundations of Artificial Intelligence: A Sourcebook*. Cambridge: CUP.

Patrick, A. S. & T. E. Whalen (1992) "Field Testing a Natural Language Information System: Usage Characteristics and Users' Comments," *Interacting with Computers* 4 (2): 218-30.

Patil, Ramesh S. et al. (1992) The DARPA Knowledge Sharing Effort: Progress Report. In Nebel, Rich & Swartout (eds.): 777-88.

Pau, L. F. & C. Gianotti (1990) *Economic and Financial Knowledge-Based Processing*. Berlin: Springer-Verlag.

Paulson, W. (1989) "Computers, Minds, and Texts: Preliminary Reflections," *New Literary History* 20: 291-303.

Pavic, M. (1988) *Dictionary of the Khazars: A Lexicon Novel in 100,000 Words*. Trans. Christina Pribicevic-Zoric. New York: Knopf.

Perelman, C. (1979) *The New Rhetoric and the Humanities*. Dordrecht, Holland: D. Reidel.

Politakis, P. G. & S. M. Weiss (1984) "Using Empirical Analysis to Refine Expert System Knowledge Bases," *Artificial Intelligence* 22: 23-48.

Poulymenakou, A., T. Cornford & E. A. Whitley (1990) "Knowledge Acquisition to Facilitate Organisational Problem Solving," *Working Paper Series No. 33*, Information Systems Dept., London School of Economics and Political Science, London.

Rada, R. (1990) "Expertext: intelligent hypertext," *Intelligent Tutoring Media* 1 (2): 55-64.

Reisberg, D. (1987) "External Representations and the Advantages of Externalizing One's Thoughts", *Proc. of the 9th Annual Cognitive Science Society*, Hillsdale, NJ: Lawrence Erlbaum, 281-93.

Rescher, N. (1979) *Cognitive Systematisation*. Oxford: Basil Blackwell.

Riesbeck, C. K. (1984) "Knowledge Reorganization and Reasoning Style," *Int. Journ. Man-Machine Studies* 20: 45-61.

Rommetveit, R. (1988) "On Human Beings, Computers, and Representational-Computational versus Hermeneutical-Dialogical Approaches to Human Cognition and Communication," Ed. Sinding-Larsen 47-76.

Salomon, G. (1979) "Media and Symbol Systems as Related to Cognition and Learning," *Journal of Educational Psychology* 71 (2): 131-48.

Salomon, G. (1987) "The Use of Visual Media in the Service of Enriching Mental Thought Processes," Eds. Boyd-Barrett & Braham 251-65.

Sampson, G. (1985) *Writing Systems*. London: Hutchinson.

Sandberg, J. (1991) "Invited Speaker: Bill Clancey," *AI Communications* 4 (1): 4-10.

Schmandt-Besserat, D. (1981) "From Tokens to tablets: A Re-evaluation of the so-called 'Numerical Tablets'," *Visible Language* 15 (4): 321-44.

Schmandt-Besserat, D. (1986) "Tokens: Facts and Interpretation," *Visible Language* 20 (3): 250-72.

Schnase, J. L. & J. J. Leggett (1989) "Computational Hypertext in Biological Modelling," *Proc. Hypertext '89*, 181-97.

Schuler, W. & J. B. Smith (1990) "Author's Argumentation Assistant (AAA): A Hypertext-Based Authoring Tool for Argumentative Texts," Eds. Streitz, Rizk & Andre: 137-51.

Schwabe, D., B. Feijo & W. G. Krause (1990) "Intelligent Hypertext for Normative Knowledge in Engineering," Eds. Streitz, Rizk & Andre 123-36.

Searle, J. R. (1990) "Minds, Brains and Programs," Ed. Boden, M. 67-87.

Sergot, M.J. et al. (1986) "The British Nationality Act as a Logic Program," *Communications of the ACM* 29 (5): 370-86.

Simon, H.A. (1986) "The Information Processing Explanation of Gestalt Phenomena," *Computers in Human Behaviour* 2: 241-55.

Simon, L. & M. Duell (1992) "A situation-related, mnemonic tool for expert tasks," *Proc. 4th Workshop on Hypertext and AI,* September, Munich. Available from FORWISS, Am Weichselgarten 7, Erlangen-Tennenlohe, Germany.

Sinding-Larsen, H. (1987) "Information Technology and the Management of Knowledge," *AI and Society* 1: 93-101.

Sinding-Larsen, H., ed. (1988) *Artificial Intelligence and Language: Old Questions in a New Key.* Oslo: Tano.

Sinding-Larsen, H. (1988a) "Information Technology and the Externalisation of Knowledge," Ed. Sinding-Larsen 77-89.

Sinding-Larsen, H. (1988b) "Notation and Music: the History of a Tool of Description and its Domain to be Described," Ed. Sinding-Larsen 91-114.

Slaney, M. (n.d.) "Lyon's Cochlear Model," *Apple Technical Report #13,* Apple Computer, Cupertino, CA.

Slaughter, M. M. (1982) *Universal Languages and Scientific Taxonomy in the Seventeenth Century.* Cambridge: CUP.

Smolesky, P. et al. (1987) "Constraint-Based Hypertext For Argumentation," *Report No. CU-CS-379-87,* Dept. of Computer Science and Institute of Cognitive Science, University of Colorado, Boulder, CO.

Smolensky, P. et al. (1988) "Computer-Aided Reasoned Discourse or, How to Argue with a Computer," Ed. Guindon, R. 109-62.

Sokolowski, R. (1988) "Natural and Artificial Intelligence," *Daedalus* 117 (1): 45-64.

Soloway, E. (1991) "Where are the New Metaphors?" *AI Communications* 4 (4): 131.

Sørgaard, P. (1991) "Evaluating Expert System Prototypes," *AI and Society* 5: 3-17.

Sowa, J. F. (1983) *Conceptual Structures: Information Processing in Mind and Machine.* Reading, Mass.: Addison-Wesley.

Steels, L. (1992) Reusability and Configuration of Applications by Non-Programmers. *VUB AI Memo 92-4,* VUB AI Lab, Pleinlaan 2, 1050 Brussels.

Stefik, M. (1986) "The Next Knowledge Medium," *AI Magazine* 7(1): 34-46.

Stefik, M. et al. (1987) "Beyond the Chalkboard: Computer Support for Collaboration and Problem Solving in Meetings," *Comm. of the ACM* 30. (1): 32-47.

Stefik, M. & L. Conway (1982) "Toward the Principled Engineering of Knowledge," *Memo KB-VLSI-82-18,* Xerox PARC, Palo Alto, CA.

Stock, B. (1984/5) "Medieval Literacy, Linguistic Theory, and Social Organization," *New Literary History* 16: 13-29.

Stonier, T. (1984) "The Knowledge Industry," Ed. Forsyth, R. 211-26.

Streitz, N. A., J. Hanneman & M. Thüring (1989) "From Ideas and Arguments to Hyperdocuments: Travelling through Activity Spaces," *Arbeitspapiere der GMD 402,* Gesellschaft für Mathematik und datenverarbeitung mbH, Schloss Birlinghoven, Postfach 1240, D-5205 Sankt Augustin 1, FRG.

Streitz, N., A. Rizk & J. Andre (1990) *Hypertext: Concepts, Systems and Applications.* Cambridge: CUP.

Sturrock, J. (1986) *Structuralism.* London: Paladin.

Stutt, A. (1989) "Argument in the Humanities: A Knowledge-Based Approach," *Technical Report No. 49,* Human Computer Research Laboratory, The Open University, Milton Keynes.

Stutt, A. (1990) "Argument Support Programs: Machines for Generating Interpretations," Ed. Miall, D. 71-81.

Tannen, D., ed. (1982) *Spoken and Written Language.* Advances in Discourse Processes, Vol. 9. Norwood, New Jersey: ABLEX Publishing.

Toulmin, S., R. Rieke & A. Janik (1979) *An Introduction to Reasoning.* New York: Macmillan Publishing.

Turner, M. (1987) *Death is the Mother of Beauty.* London: University of Chicago Press.

Troll, D. A. (1990) "The Illiterate Mode of Written Communication: The Work of the Medieval Scribe," Ed. Enos R. 96-125.

Twine, N. (1984) "The Adoption of Punctuation in Japanese Script," *Visible Language* 18 (3): 229-37.

Van De Riet, R. P. (1987) "Problems with Expert Systems?" *Future Generation Computer Systems* 3: 11-16.

Van Dyke, C. (1987) "Taking 'Computer Literacy' Literally," *Communications of the ACM* 30 (5): 366-74.

Van Nostrand, A. D. (1979) "Writing and the Generation of Knowledge," *Social Education* 43: 178-80.

Vygotsky, L. S. (1962) *Thought and Language*. Ed. & Trans. E. Hanfmann & G. Vakar, Cambridge, Mass.: MIT Press.

Waller, R. (1988) "The Typographic Contribution to Language," PhD Thesis, University of Reading.

Waterworth, J. A. & M. H. Chignall (1989) "Manifesto for Hypermedia Usability Research," *Hypermedia* 1(3): 205-33,

Wellsch, K. & M. Jones (1986) "Computational Analogy," *Proc. 7th European Conf. on Art. Int.* 21st-25th July, Brighton, UK 153-62.

West, D. M. & L. E. Travis. (1991a) "The Computational Metaphor and Artificial Intelligence: A Reflective Examination of a Theoretical Falsework," *AI Magazine* Spring 1991: 64-79.

West, D. M. & L. E. Travis (1991b) "From Society to Landscape: Alternative Metaphors for Artificial Intelligence," *AI Magazine*, Summer 1991: 69-83.

Whitaker, R. & O. Östberg (1988) "Channeling Knowledge: Expert Systems as Communications Media," *AI & Society* 2 (3): 197-208.

Winograd, T. (1990) "Thinking Machines: Can There Be? Are We?" Eds. Partridge & Wilks 167-89.

Winograd, T. & F. Flores (1986) *Understanding Computers and Cognition: A New Foundation for Design*. Norwood, New Jersey: Ablex.

Wolfram, S. (1988) *Mathematica: A System for Doing Mathematics by Computer*. Redwood City, CA: Addison-Wesley.

Wolstenholme, E. F. (1983) "Modelling National Development Programmes — An Exercise in System Description and Qualitative Analysis Using System Dynamics," *Journal of the Operational Research Society* 34 (12): 1133-48.

Woods, D. D. (1986) "Cognitive Technologies: The Design of Joint Man-Machine Cognitive Systems," *AI Magazine* 6 (4): 86-92.

Yakemovic, K.C. B. & J. Conklin (1989) "The Capture of Design Rational on an Industrial Development Project: Preliminary Report," *MCC Technical Report Number STP-279-89*. MCC, Austin, Texas.

Yankelovich, N., N. Meyrowitz & A. Van Dam (1985) "Reading and Writing the Electronic Book," *Computer* October: 15-30.

Yates F. A. (1984) *The Art of Memory*, London: Ark.

Zellweger, P T (1989) "Scripted Documents: A Hypermedia Path Mechanism," *Proc. Hypertext '89* : 1-14.